



Escola d'Enginyeria de Telecomunicació i
Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Deep learning para el reconocimiento facial de emociones básicas

TITULO DEL TFG: Deep learning para el reconocimiento facial de emociones básicas

TITULACIÓN: Grado en Ingeniería de Sistemas de Telecomunicaciones

AUTOR: Adrián Sáez de la Pascua

DIRECTOR: Esther Salamí San Juan

DATA: 31 de Enero del 2019

Título: Deep learning para el reconocimiento facial de emociones básicas

Autor: Adrián Sáez de la Pascua

Director: Esther Salamí San Juan

Fecha: 31 de Enero de 2019

Resumen

Tanto Machine learning como Deep learning son dos conceptos que en estos tiempos están presentes en muchos ámbitos, ya sea en la sanidad para los diagnósticos médicos, como en el márketing a la hora de segmentar el mercado o en la mayoría de redes sociales. En este proyecto veremos que pueden ofrecer a nivel de reconocimiento de expresiones faciales en fotos.

En la actualidad, hay una infinidad de datos que no son utilizados ya que el volumen de datos es tan grande que las personas no somos capaces de procesar tal cantidad. El Machine learning y el Deep learning ofrecen varios mecanismos para poder procesar estos datos, ordenarlos para que se pueda obtener información y sacar modelos predictivos para poder aplicarlos a otros datos.

El reconocimiento de expresiones faciales es una parte de la inteligencia artificial que tiene como objetivo principal reconocer formas básicas de expresión afectiva que aparece en los rostros de las personas. ¿Pero pueden ofrecer el Machine learning y Deep learning técnicas para poder reconocer estas expresiones faciales suficientemente eficientes?

En este proyecto aplicaremos estos mecanismos que nos ofrecen estas dos tecnologías para poder reconocer sentimientos o emociones en personas. Para ello, primero procesaremos y obtendremos la información que queremos de una base de datos. Esta base de datos son una serie de imágenes de personas mostrando diferentes sentimientos. Una vez obtenidos estos datos veremos que, aunque las dos técnicas son prometedoras en Deep learning se obtendrán mejores resultados en cuanto precisión y más robustez en el código.

Title: Deep learning for facial recognition of basic emotions

Author: Adrián Sáez de la Pascua

Director: Esther Salamí San Juan

Date: 31 of January of 2019

Overview

Both Machine learning and Deep learning are two concepts that are present in many areas, in health for medical diagnoses, as well as in marketing when segmenting the market or as in most social networks. In this project, we will see what they can offer at the level of recognition of facial expressions in photos.

At present, there is infinity of data that is not used since the volume of data is so large that people are not able to process such quantity. The Deep learning and the Machine learning offer several mechanisms to be able to process this data, order them so that they can obtain information and take predictive models to be able to apply them to other data.

The recognition of facial expressions is a part of artificial intelligence whose main objective is to recognize basic forms of affective expression that appear on the faces of people. However, can machine learning and deep learning techniques offer to be able to recognize these facial expressions sufficiently efficiently?

In this project, we will apply these mechanisms offered by these two technologies to recognize feelings or emotions in people. To do this, we will first process and obtain the information we want from a database. This database is a series of images of people showing different feelings. Once obtained these data we will see that although the two techniques are promising, in Deep learning will obtain better results in precision and more robustness in the code.

Dedicatoria

Me gustaría dedicar unas palabras a todas aquellas personas que han estado conmigo en esta etapa de mi vida. Primero de todo a Esther Salamí San Juan por atender cualquier duda, por su paciencia y guiarme en todo el proceso. En segundo lugar, a todos mis compañeros que me han acompañado durante la carrera y me han ayudado a lo largo de las asignaturas. Por último, dar gracias por el apoyo incondicional de mi familia que sin ellos no habría conseguido llegar tan lejos.

INDICE

CAPÍTULO 1. INTRODUCCIÓN	11
1.1. Motivación.....	11
1.2. Objetivos del proyecto.....	11
1.3. Organización del proyecto	12
CAPÍTULO 2. MACHINE LEARNING Y DEEP LEARNING	13
2.1. Machine learning	13
2.1.1. Redes neuronales	14
2.2. Deep learning.....	16
2.2.1. Redes neuronales convolucionales.....	17
CAPÍTULO 3. RECONOCIMIENTO DE SENTIMIENTOS	21
3.1. Herramientas utilizadas	21
3.2. Métricas para la evaluación	22
3.3. Reconocimiento de sentimientos con Machine learning.....	23
3.3.1. Recolectar datos.....	23
3.3.2. Preparar los datos	25
3.3.3. Clasificadores a evaluar	27
3.3.4. Selección del modelo	29
3.3.5. Evaluación	32
3.4. Reconocimiento de sentimientos con Deep learning	33
3.4.1. Recolectar los datos	34
3.4.2. Preparar los datos	34
3.4.3. Descripción de los modelos.....	35
3.4.4. Selección del modelo	36
3.4.5. Evaluación	38
3.5. Comparación	41
CAPÍTULO 4. CONCLUSIONES Y TRABAJO FUTURO.....	42
4.1. Conclusiones.....	42
4.2. Trabajo futuro	43
ANEXO 1: MATRICES DE CONFUSIÓN EN MACHINE LEARNING	44
ANEXO 2: MATRICES DE CONFUSIÓN EN DEEP LEARNING	53
ANEXO 3: CÓDIGO MACHINE LEARNING.....	56
ANEXO 4: CÓDIGO DEEP LEARNING	57

GLOSARIO	60
BIBLIOGRAFIA	61

INDICE DE FIGURAS

Figura 2.1 Aprendizaje supervisado	14
Figura 2.2 Aprendizaje no supervisado	14
Figura 2.3 Red neuronal.....	15
Figura 2.4 Aprendizaje profundo	16
Figura 2.5 Convolución [11].....	18
Figura 2.6 Proceso de agrupación o reducción [11]	19
Figura 2.7 Proceso conjunto de capas [12]	19
Figura 3.1 Puntos faciales [20]	24
Figura 3.2 Imagen con puntos faciales.....	25
Figura 3.3 Desplazamiento de los puntos al eje 0,0.....	26
Figura 3.4 Support Vector Machine [22]	27
Figura 3.5 Decision Tree	28
Figura 3.6 Random Forest [25].....	28
Figura 3.7 Multilayer perceptron [27]	29
Figura 3.8 Cross – Validation [28]	29
Figura 3.9 Modelos redes neuronales	36
Figura 3.10 Curva de precisión y pérdidas 2 emociones modelo 1	37
Figura 3.11 Curva de precisión y pérdidas 2 emociones modelo 2	37
Figura 3.12 Curva de precisión y pérdidas 2 emociones modelo 3	38
Figura 3.13 Curva de precisión y pérdidas 4 emociones modelo 3	39
Figura 3.14 Curva de precisión y pérdidas 7 emociones modelo 3	40

INDICE DE TABLAS

Tabla 3.1 Matriz de confusión	22
Tabla 3.2 Precisión obtenida sin estandarizar los datos	30
Tabla 3.3 Precisión obtenida estandarizando los puntos	31
Tabla 3.4 Mejores resultados	31
Tabla 3.5 Matriz de confusión MLP 2 Emociones	32
Tabla 3.6 Matriz de confusión MLP 4 Emociones	32
Tabla 3.7 Matriz de confusión MLP 7 Emociones	33
Tabla 3.8 Matriz de confusión 2 Emociones Modelo 3	39
Tabla 3.9 Matriz de confusión 2 Emociones Modelo 3	39
Tabla 3.10 Matriz de confusión 7 Emociones Modelo 3	40
Tabla 3.11 Comparación de resultados finales	41
Tabla A1.1 CM 2 Emociones SVM Sin escalar	44
Tabla A1.2 CM 2 Emociones DT Sin escalar	44
Tabla A1.3 CM 2 Emociones RF Sin escalar	44
Tabla A1.4 CM 2 Emociones MLP Sin escalar	44
Tabla A1.5 CM 2 Emociones SVM Escalado	44
Tabla A1.6 CM 2 Emociones DT Escalado	45
Tabla A1.7 CM 2 Emociones RF Escalado	45
Tabla A1.8 CM 2 Emociones MLP Escalado	45
Tabla A1.9 CM 2 Emociones SVM Estandarizado	45
Tabla A1.10 CM 2 Emociones DT Estandarizado	45
Tabla A1.11 CM 2 Emociones RF Estandarizado	45
Tabla A1.12 CM 2 Emociones MLP Estandarizado	46
Tabla A1.13 CM 4 Emociones SVM Sin escalar	46
Tabla A1.14 CM 4 Emociones DT Sin escalar	46
Tabla A1.15 CM 4 Emociones RF Sin escalar	46
Tabla A1.16 CM 4 Emociones MLP Sin escalar	47
Tabla A1.17 CM 4 Emociones SVM Escalado	47
Tabla A1.18 CM 4 Emociones DT Escalado	47
Tabla A1.19 CM 4 Emociones RF Escalado	47
Tabla A1.20 CM 4 Emociones MLP Escalado	48
Tabla A1.21 CM 4 Emociones SVM Estandarizado	48
Tabla A1.22 CM 4 Emociones DT Estandarizado	48
Tabla A1.23 CM 4 Emociones RF Estandarizado	48
Tabla A1.24 CM 4 Emociones MLP Estandarizado	49
Tabla A1.25 CM 7 Emociones SVM Sin escalar	49
Tabla A1.26 CM 7 Emociones DT Sin escalar	49
Tabla A1.27 CM 7 Emociones RF Sin escalar	49
Tabla A1.28 CM 7 Emociones MLP Sin escalar	50
Tabla A1.29 CM 7 Emociones SVM Escalado	50
Tabla A1.30 CM 7 Emociones DT Escalado	50
Tabla A1.31 CM 7 Emociones RF Escalado	51
Tabla A1.32 CM 7 Emociones MLP Escalado	51
Tabla A1.33 CM 7 Emociones SVM Estandarizado	51
Tabla A1.34 CM 7 Emociones DT Estandarizado	52
Tabla A1.35 CM 7 Emociones RF Estandarizado	52
Tabla A1.36 CM 7 Emociones MLP Estandarizado	52
Tabla A2.1 CM 2 Emociones Modelo 1	53

Tabla A2.2 CM 2 Emociones Modelo 2	53
Tabla A2.3 CM 2 Emociones Modelo 3	53
Tabla A2.4 CM 4 Emociones Modelo 1	53
Tabla A2.5 CM 4 Emociones Modelo 2	53
Tabla A2.6 CM 4 Emociones Modelo 3	54
Tabla A2.7 CM 7 Emociones Modelo 1	54
Tabla A2.8 CM 7 Emociones Modelo 2	54
Tabla A2.9 CM 7 Emociones Modelo 3	55

CAPÍTULO 1. INTRODUCCIÓN

1.1. Motivación

La inteligencia artificial ha tenido una implicación cada vez mayor en cualquier ámbito de la vida del ser humano. Las tecnologías se adaptan a las necesidades del ser humano y la inteligencia artificial es la que hace posible esta adaptación entre tecnología y ser humano.

La aparición de arquitecturas de Big Data ha ido muy ligada a grandes compañías de internet y a muchas organizaciones de todos los sectores, ya que todas necesitan pasarse a la transformación digital. En la actualidad, el Machine learning y el Deep learning son dos conceptos que las grandes multinacionales hacen uso de ellos para poder adaptarse mejor a las necesidades del día a día, desde las empresas dedicadas a la sanidad para poder tener un mayor conocimiento del paciente y así ofrecer un mejor diagnóstico, hasta las empresas dedicadas a la conducción autónoma para ofrecer una mayor seguridad a los ocupantes del vehículo.

Estas técnicas están muy presentes en algoritmos de reconocimiento de expresiones humanas. Cuando el ser humano intenta comunicarse con otras personas un porcentaje muy alto es representado por la comunicación no verbal. Muchos estudios demuestran que las expresiones faciales tienen una vinculación con las emociones humanas. La capacidad que tiene el ser humano para detectar e identificar estas emociones hace posible que nos podamos entender los unos a los otros. El objetivo principal de esta parte de la inteligencia artificial es utilizar técnicas de aprendizaje para que la máquina sea capaz de identificar estas emociones.

Hoy en día se hace uso de estas arquitecturas para todo tipo de ámbitos de la vida desde estudios sociológicos, videojuegos, en seguridad o incluso en la detección de enfermedades. A través de este estudio las grandes empresas son capaces de aplicarlo a diferentes tecnologías como por ejemplo Facebook para etiquetar mejor las fotos subidas a la red social o Microsoft que tiene una API capaz de detectar el tanto por ciento de cada emoción de cada cara en una imagen.

1.2. Objetivos del proyecto

El objetivo principal de este proyecto es investigar la utilización de técnicas de aprendizaje profundo para el reconocimiento de emociones en imágenes faciales. Para ello se plantean los siguientes objetivos parciales:

- Recopilación de una base de datos de imágenes de expresiones faciales.

- A partir de una base de datos de imágenes con diferentes expresiones faciales, se obtienen varios parámetros para la extracción de características de estas para su posterior utilización como entrada en un sistema de reconocimiento.
- Implementación y evaluación de un sistema de aprendizaje automático para el reconocimiento de expresiones basado en Python, Sklearn, OpenCV y Dlib.
- Análisis, implementación y evaluación de un sistema de aprendizaje profundo para el reconocimiento de expresiones con Keras.

1.3. Organización del proyecto

El proyecto está dividido en tres partes:

- Una primera parte introductoria a los términos de Machine learning y Deep learning para que se puedan entender cómo funcionan estos conceptos y las diferencias entre ellos.
- Una segunda parte de estudio de análisis de los sentimientos. Aquí es donde veremos las diferentes técnicas que utilizan los dos conceptos, como procesan los datos, como modificar los datos para que puedan utilizarlos correctamente y los pasos a seguir para poder realizar el estudio.
- Por último, una parte donde se podrán ver los resultados obtenidos, conclusiones obtenidas de los datos y algunos posibles futuros para estas técnicas.

CAPÍTULO 2. Machine learning y Deep learning

En la actualidad Machine learning y Deep learning están muy presentes en muchos aspectos de la vida humana. Por ejemplo, el Machine learning es muy utilizado en el Marketing a la hora de saber más datos sobre los clientes y su comportamiento, así poder ofrecer los productos que necesitan y se traducirá en más ventas [1]. Para aprendizaje profundo, está muy presente en la traducción inteligente como por ejemplo el de Google [2]. Así que en esta parte se procederá a explicar la base de estos dos conceptos.

2.1. Machine learning

El Machine learning, conocido en español como aprendizaje automático o aprendizaje de máquina, pretende estudiar el reconocimiento de patrones y el aprendizaje por parte de las computadoras. Hace posible que las máquinas puedan aprender sin ser explícitamente programadas.

Esta disciplina trabaja con algoritmos que pueden dar hallazgos o conclusiones relevantes obtenidas de un conjunto de datos, sin que el ser humano tenga que escribir instrucciones o códigos para esto. El propósito de esta disciplina es que las personas y las máquinas trabajen de la mano. Precisamente esto es lo que hacen los algoritmos, permiten que las máquinas ejecuten tareas, tanto generales como específicas.

El principal objetivo del aprendizaje automático es desarrollar la capacidad de generalizar y asociar, se podría traducir en que las máquinas deberían poder desempeñarse con precisión y exactitud, tanto en tareas familiares, como en actividades nuevas o imprevistas.

Esta técnica imita la capacidad del ser humano para procesar la información y así poder crear modelos que generalicen la información obtenida para realizar las predicciones.

Aunque hay muchas categorías, se ha centrado este proyecto en dos categorías: aprendizaje supervisado y aprendizaje no supervisado [3].

- **Aprendizaje supervisado:** Facilita buscar patrones de datos y relacionarlos a un campo llamado objetivo [4]. Se entrena al algoritmo usando unos atributos (features) con sus correspondientes etiquetas (labels) [5]. El proyecto se centrará en esta parte del aprendizaje.

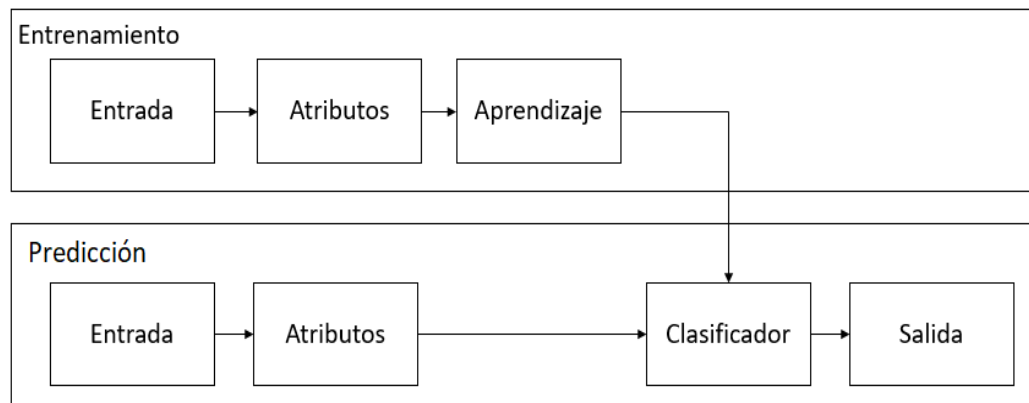


Figura 2.1 Aprendizaje supervisado

- **Aprendizaje no supervisado:** Son aquellos sistemas en los que no disponemos de una base de datos de ejemplos que ya han sido clasificados, sino que a partir de las propiedades de los ejemplos intentamos dar una clasificación de los ejemplos según su similitud [6].

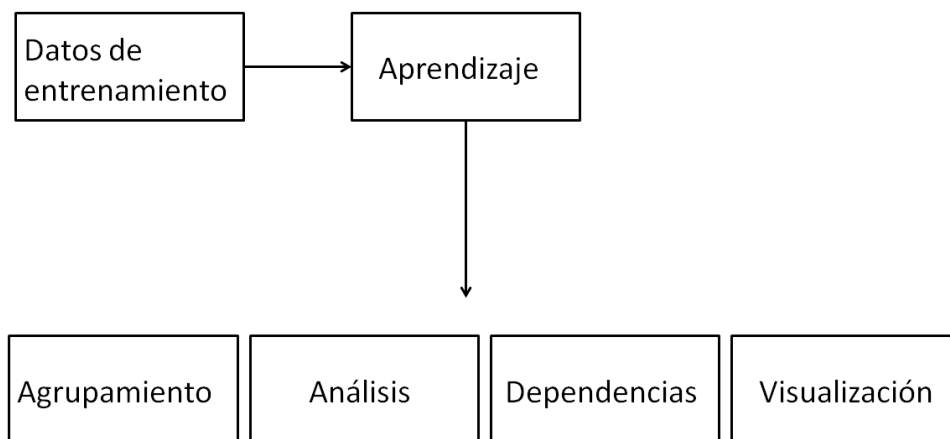


Figura 2.2 Aprendizaje no supervisado

2.1.1. Redes neuronales

El cerebro humano se compone de más de cien millones de redes neuronales. Estas redes son las que hacen que podamos aprender y recordar información, así como el procesamiento de dicha información.

Las redes neuronales son sistemas de programas y estructuras que intentan imitar el funcionamiento del cerebro humano [7]. Son sistemas que enlazan las neuronas para que puedan trabajar entre ellas y así producir un estímulo de salida. Así son capaces de aprender y pueden adaptarse a las diferentes entradas que les pueden llegar.

Las redes neuronales constan de capas de entrada, capas ocultas y capas de salida.

- **Capa de entrada:** Recibe los datos de entrada y los pasa a la primera capa oculta. Esta capa tendrá tantas neuronas como entradas tenga la red.
- **Capa oculta:** Realizarán los cálculos matemáticos con nuestras entradas. Las redes neuronales pueden estar compuestas por varias de estas capas.
- **Capa de salida:** Una vez que la red realiza las operaciones matemáticas, contiene el resultado. Habrá tantas neuronas como clasificaciones haya.

Esquema de capas en una Red Neuronal

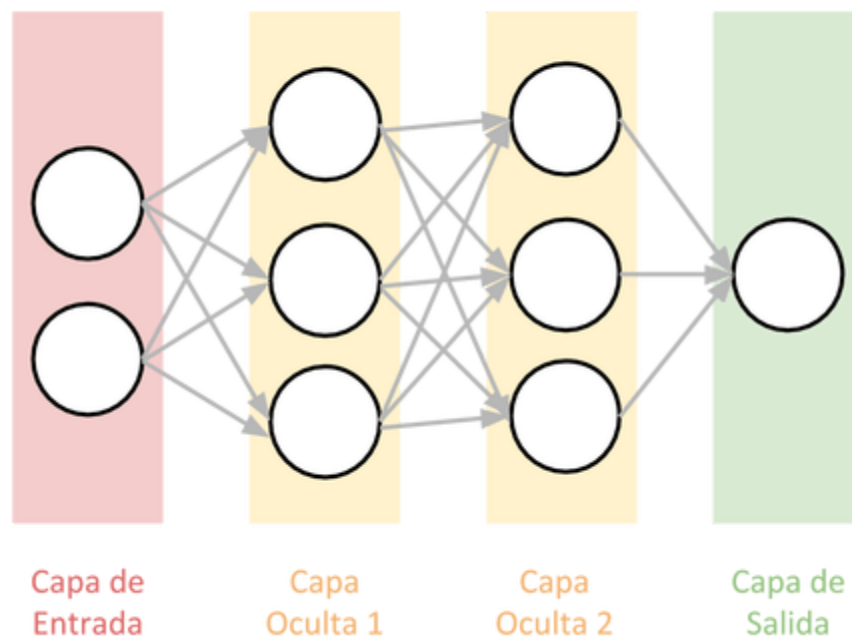


Figura 2.3 Red neuronal

Como se puede ver en la Figura 2.4 todas las neuronas están interconectadas entre ellas con las diferentes capas. Cuanto mayor es el número de capas, más profunda es la red neuronal [8].

En la mayoría de los casos, todas las neuronas de cada capa tienen una conexión con cada una de las neuronas de la siguiente capa. Cada conexión tiene asociado un peso. El peso determina la importancia que tendrá esa conexión con la neurona. Estos pesos tienen un valor inicial que se asigna aleatoriamente. La principal operación que realiza la red es multiplicar los valores de una neurona por los pesos de sus conexiones salientes. Cada neurona de la siguiente capa recibe números de varias conexiones entrantes y los suma [9].

2.2. Deep learning

Deep learning, conocido en español como aprendizaje profundo, son aquellas técnicas de aprendizaje automático que hacen uso de arquitecturas de redes neuronales. El concepto “profundo” viene referido al número de capas que poseen estas técnicas. Mientras que las redes neuronales convencionales poseen una capa, las redes neuronales profundas contienen varias capas.

Aprendizaje profundo es un aspecto de la inteligencia artificial que se ocupa de emular el enfoque de aprendizaje que utilizamos los seres humanos para obtener ciertos tipos de conocimiento. Puede considerarse una forma de automatizar el análisis predictivo.

Los algoritmos de aprendizaje profundo se apilan en una jerarquía de creciente complejidad y extracción. Cada algoritmo en la jerarquía aplica una transformación no lineal en su entrada y utiliza lo que aprende para crear un modelo estadístico como salida. Las iteraciones continúan hasta que la salida ha alcanzado un nivel de precisión aceptable [10].

En aprendizaje profundo el modelo aprende por sí mismo y poder así descubrir las relaciones entre las variables. En los modelos basados en aprendizaje profundo no interviene el ser humano, el propio modelo es capaz de hacer la selección de características.

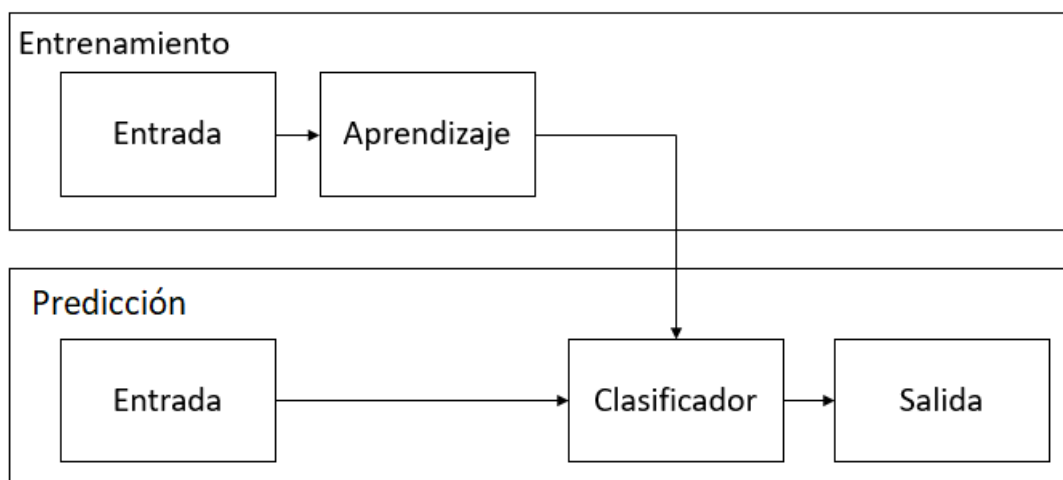


Figura 2.4 Aprendizaje profundo

En este capítulo nos centraremos en la arquitectura de aprendizaje profundo conocida como Convolutional Neural Network (CNN), conocida en español como Redes Neuronales Convolucionales, ya que este tipo de redes son muy efectivas para tareas de detección y clasificación de imágenes.

2.2.1. Redes neuronales convolucionales

Como ya se ha comentado anteriormente estas técnicas hacen uso de redes neuronales, pero en aprendizaje profundo se utilizan unas redes neuronales especiales llamadas redes neuronales convolucionales. Estas redes son muy similares a las redes neuronales, están formadas por neuronas que tienen forma de pesos y sesgos que se pueden aprender. Están compuestas por tres capas (explicadas anteriormente) pero existen unas capas ocultas que están especializadas en dos operaciones: convolución y agrupación.

2.2.1.1. Capa de convolución

Estas capas ocultas se ocupan de aprender patrones locales en ventanas pequeñas de dos dimensiones, alto por ancho. Son capas que sirven para detectar características visuales en las imágenes como líneas, bordes y gotas de color. Estas capas permiten aprender una propiedad de la imagen en un punto específico y ser capaz de reconocerla en cualquier lugar de la propia imagen.

Las capas convolucionales son capaces de aprender diferentes elementos más complejos en función de lo aprendido en la capa anterior. Un ejemplo sería que una capa aprenda diferentes tipos de líneas que aparezcan en la imagen y la capa siguiente sea capaz de aprender elementos de la imagen que estén compuestos por diferentes líneas aprendidas en la capa anterior. La capacidad de realizar esto permite a estas redes poder aprender de una manera eficiente conceptos visuales que cada vez son más complejos.

Las redes neuronales convolucionales operan sobre 3 ejes, ancho, alto y otro eje que sería el canal. Este canal podría ser de 1 si la imagen de entrada es en una escala de grises o bien 3 para una imagen en color RGB, una por cada color (rojo, verde y azul).

No todas las neuronas de entrada estarán conectadas con todas las neuronas de esta capa de convolución. Se realiza mediante pequeñas áreas localizadas del espacio de las neuronas de entrada que almacenan los píxeles de la imagen.

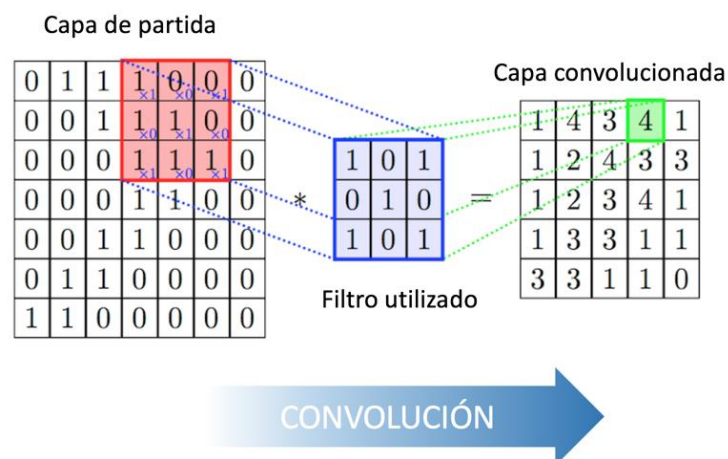


Figura 2.5 Convolución [11]

Como se puede observar en la Figura 2.6, esta matriz llamada Kernel recorre todas las neuronas de entrada. Por cada posición de la matriz hay una neurona.

La matriz empieza a recorrer todas las neuronas por la parte superior izquierda y se desplaza hacia la derecha. Cuando ha acabado una fila, sigue con la fila de debajo. Así se consigue obtener información de todas las neuronas y que estén conectadas entre ellas.

Con este método se puede observar que se define un espacio de neuronas a la salida inferior al de entrada, para ser exactos se reduce a:

$$\begin{aligned}
 &(\text{Neuronas alto} \times \text{Neuronas ancho}) \text{ salida} \\
 &= (\text{Neuronas ancho} - \text{ancho matriz} - 1) \times (\text{Neuronas alto} \\
 &\quad - \text{alto matriz} - 1) \text{ entrada}
 \end{aligned} \quad (2.1)$$

Esto puede ocasionar pérdida de información, por lo que en las redes neuronales se aplica una técnica de relleno de ceros para no perder neuronas.

Esta capa de convolución aplica una serie de filtros donde cada filtro detecta una propiedad de la imagen. Por lo que a la salida se podrá observar un tamaño de:

$$\text{Neuronas salida} = \text{Neuronas ancho} \times \text{Neuronas alto} \times \text{filtros} \quad (2.2)$$

2.2.1.2. Capa de agrupación o reducción

A continuación de las capas convolucionales, las redes están acompañadas por unas capas de agrupación, también llamadas capas de reducción. Como su nombre indica, agrupa las neuronas en grupos. Este proceso ayuda a obtener

las propiedades más relevantes de los datos de entrada y obviar datos superfluos que pueden inducir a error.

Existen varias formas de agrupación. Una sería “agrupación promedio” que obtiene el valor promedio del grupo de neuronas o bien “agrupación por valor máximo” que sería agrupaciones de neuronas donde se escoge el valor máximo de cada grupo.

Esta capa, al agrupar las neuronas la entrada se ve reducida. Se reduce en función del tamaño de las agrupaciones. Por ejemplo, si los grupos son de un tamaño 2x2, a la salida obtendremos la mitad del tamaño de los datos de entrada. Estas agrupaciones también se aplican tantas veces como profundidad tengan los datos que viene determinado por el número de filtros aplicados en la capa anterior. Se puede observar que, aunque se aplique esta técnica de transformación, se sigue manteniendo la relación espacial [12].

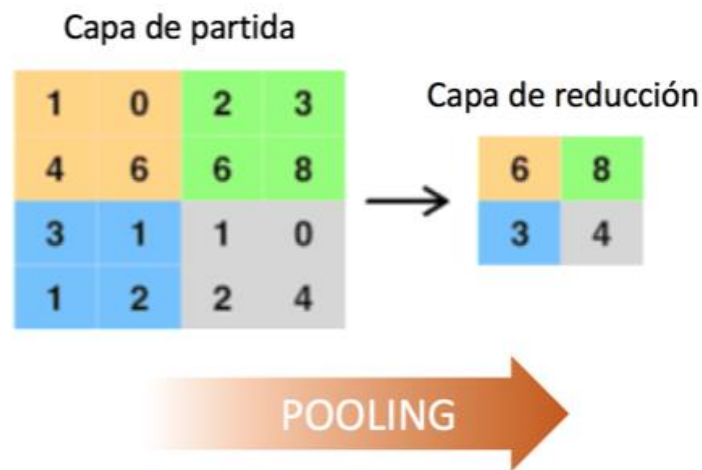


Figura 2.6 Proceso de agrupación o reducción [11]

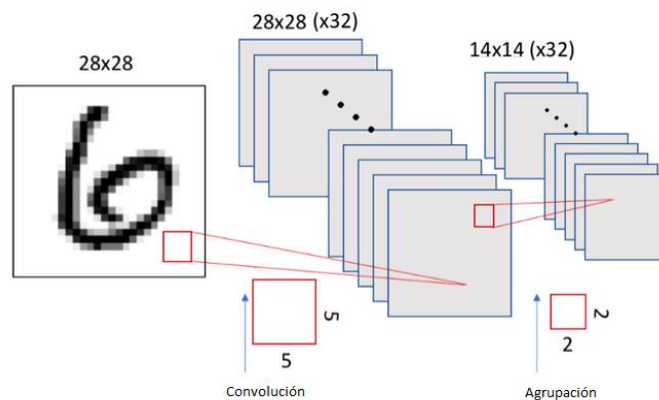


Figura 2.7 Proceso conjunto de capas [12]

La Figura 2.6 es un ejemplo de cómo se ve alterado el tamaño de la entrada por el paso de las capas. Inicialmente entra una imagen de tamaño 28x28 y se aplica una capa de convolución. En este caso no se ve alterado el tamaño ya que se utiliza la función de relleno de ceros. Sobre esta capa, se aplican un total de 32 filtros de 5 x 5 que les proporciona profundidad a los datos. Por último, se agrupan los datos en grupos de 4 por lo que se ve reducida la imagen a la mitad, pero sin perder profundidad.

CAPÍTULO 3. Reconocimiento de sentimientos

En este proyecto se centrará en el reconocimiento de expresiones faciales en las imágenes. El objetivo principal es comparar el rendimiento obtenido utilizando técnicas tradicionales de aprendizaje automático frente a técnicas de aprendizaje profundo. En la primera parte aplicaremos técnicas de aprendizaje automático utilizando como herramientas Python, Sklearn, OpenCV y Dlib, mientras que en la segunda parte del proyecto utilizaremos métodos de aprendizaje profundo utilizando como herramienta Python y Keras. El código implementado está en los Anexos 3 y 4.

3.1. Herramientas utilizadas

Para poder trabajar con técnicas de aprendizaje automático y aprendizaje profundo hemos tenido que hacer uso de una serie de herramientas. En algunas herramientas se ha especificado la versión ya que no todas las versiones son compatibles entre ellas. Las herramientas utilizadas han sido:

- **JetBrains Pycharm Community Edition:** Es el entorno de desarrollo integrado utilizado para programar (IDE). Este entorno de programación es específico para el lenguaje de Python que es el que se usará [13].
- **Python:** Es el lenguaje de programación que se utilizara para implementar estas técnicas. Se ha utilizado la versión 2.7 para la parte de aprendizaje automático y la versión 3.6 para aprendizaje profundo.
- **Numpy:** Librería de funciones matemáticas de alto nivel para que pueda operar con esos vectores o matrices. [14]
- **Pip:** Herramienta utilizada para instalar librerías de Python de una forma más cómoda, más eficiente y más rápida. Para aprendizaje automático se ha utilizado la versión 2.0 y para aprendizaje profundo la 3.0.
- **Dlib:** Librería que sirve para la extracción de los puntos faciales de cada rostro. Proporciona un código que extrae de cada imagen 68 puntos faciales de cada rostro. Las técnicas de análisis de imágenes en aprendizaje automático utilizan los 68 puntos para el reconocimiento de las expresiones faciales.
- **Open CV:** Librería libre de visión artificial originalmente desarrollada por intel [15].
- **Keras:** Librería de redes neuronales artificiales escrita en Python. Es la librería que contiene las diferentes técnicas que se harán uso en aprendizaje profundo. La versión utilizada ha sido la 2.0.8.

- **Compute Unified Device Architecture (CUDA):** Plataforma de computación en paralelo incluyendo un compilador y un conjunto de herramientas de desarrollo creadas por NVIDIA que permite usar una variación del lenguaje de programación C para codificar algoritmos en GPU de NVIDIA [16]. La versión utilizada es la 8.0
- **Cudnn:** Biblioteca que proporciona implementaciones altamente optimizadas para rutinas estándar tales como convolución, agrupación, normalización y capas de activación. Permite centrarse en la formación de redes neuronales y en el desarrollo de aplicaciones de software en lugar de dedicar tiempo al ajuste de rendimiento de GPU [17]. La versión utilizada ha sido la 5.1.
- **Tensorflow-gpu:** Sistema de “backend” que permite compilar código de redes neuronales profundas. Si utilizamos la extensión “GPU” permitimos al ordenador que sea capaz de procesar a través de la unidad de procesamiento gráfico de la propia tarjeta gráfica. La versión utilizada ha sido la 1.5.

3.2. Métricas para la evaluación

Para poder evaluar y ver si los modelos son realmente eficientes haremos uso de una serie de métricas:

- **Precisión:** Parámetro que proporciona cuanto de acertado ha sido ese modelo al predecir.

$$\text{Precisión} = \frac{\text{Emociones detectadas correctas}}{\text{Emociones totales}} \times 100 \quad (3.1)$$

- **Matriz de confusión:** Herramienta que permite ver cuántas predicciones se han realizado, cuales ha acertado y cuales ha errado. Solo se mostrará las matrices de confusión de los resultados finales. Los valores de las matrices se mostrarán en tanto por ciento. Las matrices de los pasos intermedios se pueden consultar en los anexos.

Tabla 3.1 Matriz de confusión

		PREDICCIÓN	
		POSITIVOS	NEGATIVOS
REALES	POSITIVOS	VERDADEROS POSITIVOS	FALSOS NEGATIVOS
	NEGATIVOS	FALSOS POSITIVOS	VERDADEROS NEGATIVOS

- Pérdidas: Parámetro que permite observar cuales de las muestras se ha perdido a causa de predicciones erróneas. Es el grado de error entre la medida calculada a la salida y la respuesta deseada a la salida. Esto es específico de aprendizaje profundo.

3.3. Reconocimiento de sentimientos con Machine learning

Para poder trabajar la primera parte del proyecto hay que seguir una serie de pasos [18]

- Recolectar los datos.
- Preparar los datos.
- Clasificadores a evaluar
- Entrenar y evaluar nuestros clasificadores

3.3.1. Recolectar datos

Para este proyecto necesitaremos obtener las expresiones faciales de todos los rostros de las imágenes, así que en esta parte veremos la BBDD que hemos utilizado, cómo funcionan las librerías de Dlib y si realmente se ajustan a nuestras necesidades, y como almacenamos la información obtenida.

Hemos obtenido una base de datos, proporcionada por [19], de 36 personas con 7 sentimientos cada una (neutral, feliz, triste, sorpresa, rabia, disgusto y miedo). Estas imágenes se han tomado sobre una superficie negra donde se puede ver claramente a la persona con el sentimiento que está mostrando. Son fotos hechas frontalmente, con una buena luminosidad, todas con la misma resolución y sin ningún ángulo de inclinación para que a la hora de recoger las expresiones faciales nos sea más fácil trabajar con ellas.

Ahora veremos como muestra las expresiones faciales la librería Dlib. Gracias al código proporcionado por [20], nos mostrará en las imágenes estas líneas faciales. El detector de líneas faciales implementado dentro del Dlib produce 68 puntos donde cada punto tiene su coordenada "X" e "Y". En la Figura 1 se muestra cómo están distribuidos estos 68 puntos.

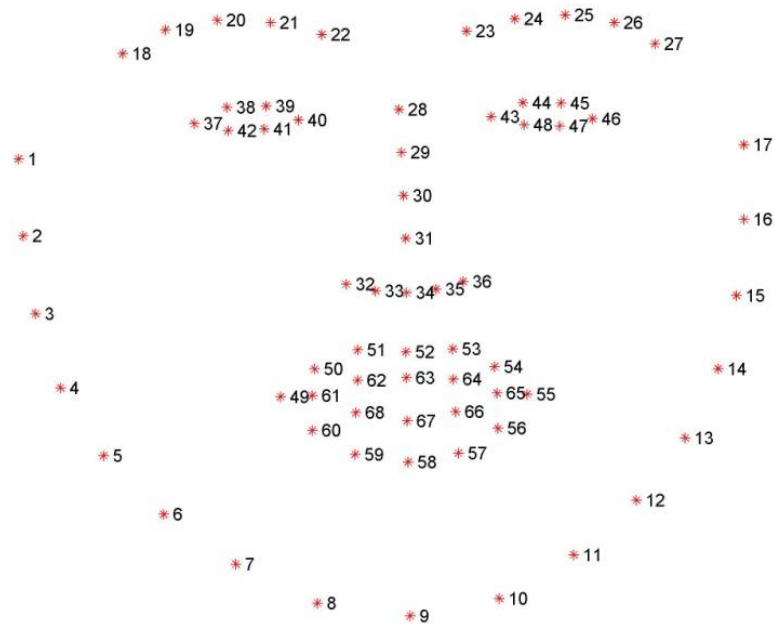


Figura 3.1 Puntos faciales [20]

Como se puede observar en la Figura 1, cada rostro constará de estos 68 puntos, donde:

- 1-17 son los puntos de la forma de la barbilla
- 18-22 son los puntos de la ceja izquierda.
- 23-27 son los puntos de la ceja derecha.
- 28-31 son los puntos de la nariz.
- 32-36 son los puntos de la parte de debajo de la nariz.
- 37-42 son los puntos del ojo izquierdo.
- 43-48 son los puntos del ojo derecho.
- 39-68 son los puntos de la boca.

En la Figura 2 se muestra como muestra los puntos la librería del Dlib:

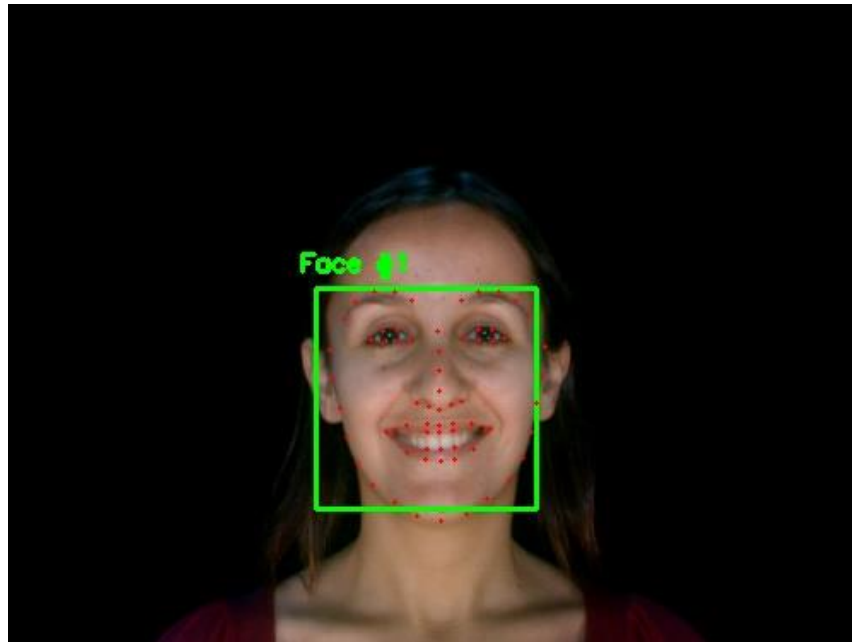


Figura 3.2 Imagen con puntos faciales

En las imágenes se pueden ver bien los 68 puntos que detecta el Dlib. También muestra en un cuadrado verde las diferentes caras detectadas en la imagen.

Estos datos han sido recogidos en un archivo “csv” para poder trabajar de una forma más fácil y así poderlos utilizar en los diferentes clasificadores. Cada línea de este archivo correspondería a las coordenadas “X” e “Y” de las líneas de expresión de cada rostro de la BBDD. Las filas de este archivo corresponderán al número de rostros de la BBDD y las columnas corresponderán a las coordenadas “X” e “Y” de todas las líneas de expresión distribuidas en x_0 , y_0 , x_1 , y_1 , etc... El orden de los puntos esta detallado en la Figura 3.1. Se utilizará otro archivo “csv” en el cual se indicará a cada rostro a qué emoción corresponde. Este tendrá el mismo número de filas que el anterior archivo “csv”, pero solo tendrá una columna ya que cada cara solo tendrá una emoción.

3.3.2. Preparar los datos

Aunque nuestra BBDD ya está adecuada a nuestras necesidades, tendremos que corregir nuestros puntos para poder facilitar el trabajo a los clasificadores. Todos los puntos se encontrarán en centro de la imagen inicialmente, así que hemos trasladado todos los puntos al punto 0,0. Para ello a todos los puntos le tendremos que restar la coordenada “X” e “Y” de la esquina superior izquierda del cuadrado verde pintado por el código por defecto o si hay un punto de las líneas de expresión más pequeño que ese valor, se utilizará ese valor. Así todos los valores estarán localizados entre 0 y el punto de la expresión facial más grande.

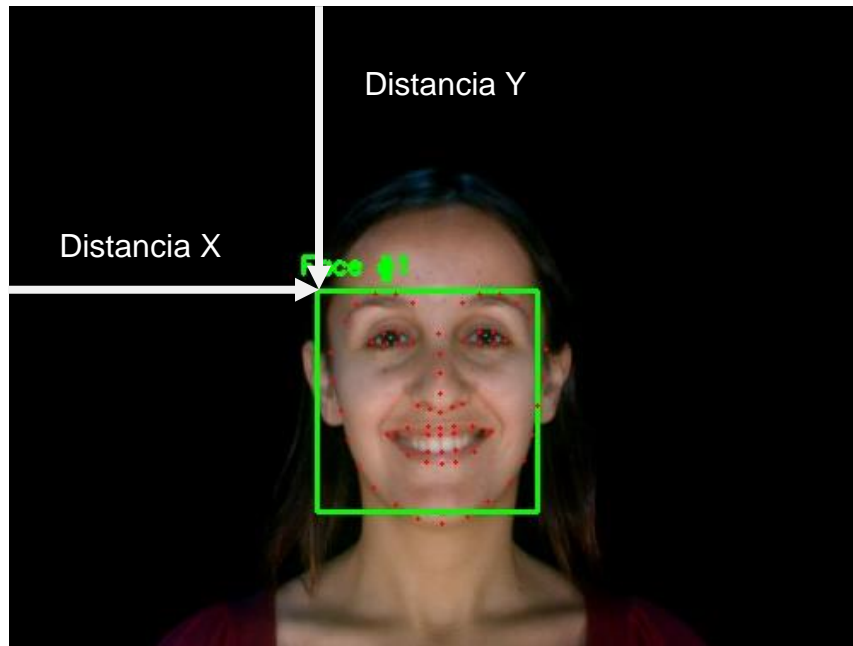


Figura 3.3 Desplazamiento de los puntos al eje 0,0

Ya que no todos los rostros tendrán el mismo tamaño, es posible que tengamos que poner a la misma escala los puntos. Para escalar los puntos, habrá que hacerlo sobre las coordenadas más grandes de las líneas de expresión o por defecto utilizará las coordenadas de la esquina inferior derecha del cuadro verde. Escalaremos también sobre la coordenada “X” o “Y” más grande.

$$X = \frac{X_{in} - X_{min}}{X_{max} - X_{min}} \quad (3.2)$$

$$Y = \frac{Y_{in} - Y_{min}}{Y_{max} - Y_{min}} \quad (3.3)$$

En la ecuación 3.2 se puede observar como son las funciones de escalado. X_{in} es la coordenada X del punto facial de entrada, X_{max} es el punto más grande de coordenada X de todos los puntos faciales, X_{min} es la coordenada más pequeña de todos los puntos faciales y X es la coordenada a la misma escala. La ecuación 3.3 es lo mismo, pero para la coordenada Y.

Aunque todos los puntos son importantes no todos sirven para determinar las emociones, así que hemos decidido no utilizar los puntos de la forma de la barbilla ya que podrían inducir a error. Cuantos menos puntos superfluos

tengamos, más precisión obtendremos a la hora de evaluar los diferentes clasificadores.

Probaremos dos casos poniendo todos los puntos a la misma escala sin estandarizar y luego estandarizando las coordenadas.

La estandarización de los conjuntos de datos es un requisito común para muchos clasificadores; podrían tener un comportamiento erróneo si las características de los datos no se parecen más o menos a los datos estándar distribuidos normalmente: gaussianos con media cero y variación de unidades [19]. Para estandarizar, se resta a cada dato la media de la variable y se divide por la desviación típica.

$$Z = \frac{X_i - X}{\delta} \quad (3.4)$$

3.3.3. Clasificadores a evaluar

Para evaluar la propuesta hemos seleccionado 4 clasificadores de Sklearn de algunas de las familias más representativas de algoritmos supervisados:

- **Support Vector Machines (SVM):** Este modelo representa los puntos de muestra en el espacio, separando las clases a 2 espacios más amplios posibles mediante un hiperplano de separación definido como el vector entre los 2 puntos [21].

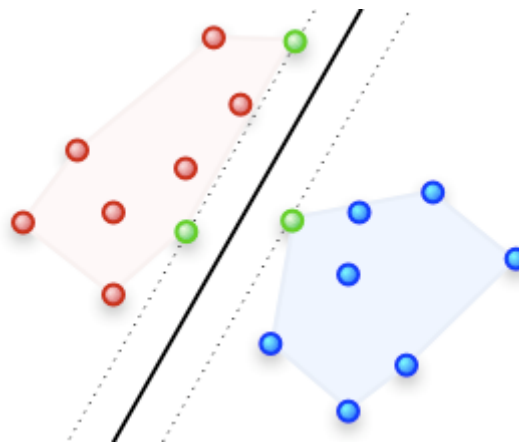


Figura 3.4 Support Vector Machine [22]

- **Decision Tree (DT):** El objetivo de este clasificador es crear un modelo que prediga el valor de un objetivo mediante el aprendizaje de reglas de decisión simples inferidas a partir de las características de los datos.

Los árboles de decisión aprenden de los datos para aproximar una curva sinusoidal con un conjunto de reglas de decisión “if-then-else” [23].

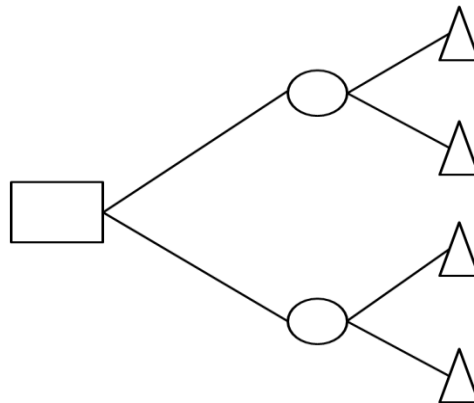


Figura 3.5 Decision Tree

- **Random Forest (RF):** Es un estimador que se ajusta a una serie de clasificadores de árboles de decisión en varias submuestras del conjunto de datos y utiliza el promedio para mejorar la precisión predictiva y el ajuste excesivo del control. El tamaño de la submuestra es siempre el mismo que el tamaño de la muestra de entrada original, pero las muestras se extraen con reemplazo [24].

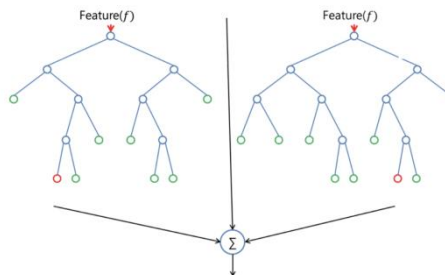


Figura 3.6 Random Forest [25]

- **Multilayer perceptron (MLP):** Es una red neuronal artificial formada por varias capas. Este clasificador puede estar totalmente conectado o no. Si está totalmente conectado cada salida de la neurona pertenece a la entrada de la siguiente capa, sino lo está cada neurona de una capa es entrada de una serie de neuronas de la siguiente capa [26].

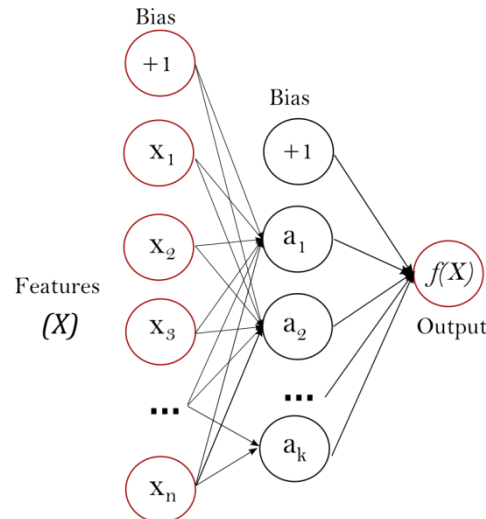


Figura 3.7 Multilayer perceptron [27]

3.3.4. Selección del modelo

Para validar los clasificadores utilizaremos una técnica conocida como “Cross-Validation”. Esta técnica consiste en repetir y calcular la media aritmética obtenida de las medidas de evaluación sobre diferentes particiones [28]

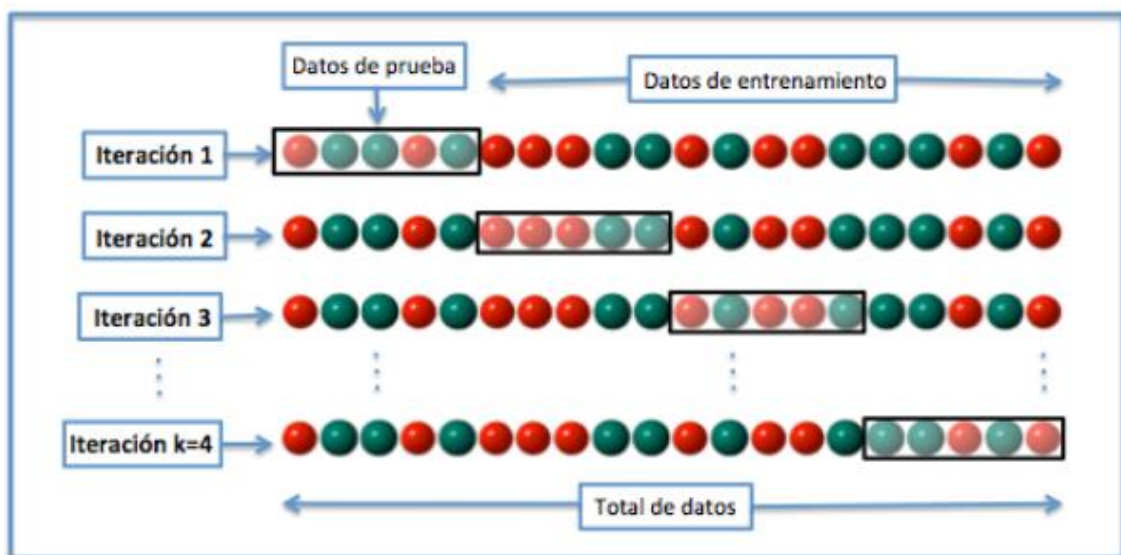


Figura 3.8 Cross – Validation [28]

Como se puede ver en la ilustración, se dividen las muestras en diferentes particiones, unas particiones se utilizan para entrenar al clasificador y las otras particiones para utilizar el clasificador y predecir. Esta prueba se ejecuta tantas veces como particiones le hagas a los datos, así que hemos cogido un valor que más comúnmente se utiliza para este método que es 10. Es una técnica

que en este caso servirá muy bien ya que entrena el mismo clasificador con diferentes imágenes e intenta predecir las otras. Esto facilita el trabajo ya que no hace necesario realizar particiones de la BBDD.

Se han hecho varios estudios con la BBDD, probando con diferentes emociones de la misma BBDD. Primero con 2 sentimientos los que correspondería a feliz y triste. Aquí se espera obtener una precisión muy alta ya que al ser dos emociones como mínimo obtendremos un 50%. A continuación probaremos con 4 sentimientos que correspondería a feliz, triste, rabia y sorpresa. Y por último evaluaremos los clasificadores sobre los 7 sentimientos (feliz, triste, rabia, miedo, sorpresa, disgusto y neutral).

Una vez elegidos los clasificadores y la forma de entrenarlos, se procede a estudiarlos sin estandarizar los datos. La Tabla 3.2 muestra la precisión que se ha obtenido con los diferentes clasificadores y variando el número de emociones.

Tabla 3.2 Precisión obtenida sin estandarizar los datos

	Puntos no estandarizados		
	2 Emociones	4 Emociones	7 Emociones
SVM	86%	61%	42%
DT	85%	62%	38%
RFC	88%	63%	41%
MLP	95%	77%	50%

Podemos ver que obtenemos mayor precisión con solo 2 sentimientos ya que sería lo más fácil de predecir. A medida que aumentamos el número de sentimientos, la precisión baja, ya que se vuelve más difícil identificar las diferentes emociones. Como mucho se obtendrá una precisión del 50% para 7 sentimientos.

El mejor clasificador en este caso es el MLP, es el que ofrece mayor precisión en los 3 casos. Para aumentar la precisión del Decisión Tree habría que aumentar el número de ramas, pero correríamos el riesgo de sobre ajustar el clasificador. Esto haría que este clasificador funcionase muy bien para esta BBDD, pero provocaría más errores en otras imágenes.

Ahora veremos cómo funcionan los clasificadores con los valores estandarizados.

Tabla 3.3 Precisión obtenida estandarizando los puntos

	Puntos estandarizados		
	2 Emociones	4 Emociones	7 Emociones
SVM	95%	77%	53%
DT	85%	62%	38%
RF	88%	63%	41%
MLP	95%	78%	56%

Como se puede observar, hemos obtenido los mejores resultados estandarizando. Mientras que SVM y MLP mejoran la precisión estandarizando los puntos, vemos que DT y RF son clasificadores no sensibles a la estandarización.

En la Tabla 3.4 se pueden observar los mejores resultados de cada clasificador y su procedimiento.

Tabla 3.4 Mejores resultados

	Mejores resultados					
	2 Emociones		4 Emociones		7 Emociones	
	Porcentaje	Estandarizado	Porcentaje	Procedimiento	Porcentaje	Procedimiento
SVM	95%	SI	77%	SI	53%	SI
DT	85%	SI/NO	62%	SI/NO	38%	SI/NO
RF	88%	SI/NO	63%	SI/NO	41%	SI/NO
MLP	95%	NO	78%	SI	56%	SI

Hemos obtenido mejor precisión estandarizando los puntos. El DT y el RF son clasificadores que no se ven afectados por la estandarización.

Cuando solo hemos utilizado la BBDD de dos sentimientos hemos obtenido hasta un 95 % de precisión. Este es un buen resultado. Al utilizar 4 sentimientos las precisiones bajan. Las líneas de decisión entre cada emoción se van ajustando lo que van haciendo que baje, pero aun así hemos obtenido un 78% en el mejor de los casos. 7 emociones es un número elevado de sentimientos, algunos sentimientos pueden ser muy parecidos (Ej: neutral y feliz) así que es normal no obtener unos niveles de precisión muy altos. La mayor precisión que hemos obtenido es un 56%. Solo acertarían poco más de la mitad de las emociones.

3.3.5. Evaluación

Como resultado final, se ha podido ver que el mejor clasificador ha sido el MLP cuando se han estandarizado las coordenadas. Aun así, hay sentimientos que le cuesta reconocer.

Tabla 3.5 Matriz de confusión MLP 2 Emociones

2 Emociones			
MULTILAYER PERCEPTRON			
	FELIZ	TRISTE	TOTAL
FELIZ	91,67%	8,33%	100%
TRISTE	2,78%	97,22%	100%
TOTAL	94,45%	105,55%	

En la Tabla 3.5 se muestra la matriz de confusión para el caso de dos emociones. Se observa que tanto para feliz como triste se obtiene más de un 90 % de precisión para estos sentimientos. La columna de la derecha se puede ver el total de imágenes reales, mientras que el total que aparece en la parte inferior muestra el total de emociones predichas. Para poder lograr una precisión del 100% solo tendría que tener un 100 % la diagonal descendente.

Tabla 3.6 Matriz de confusión MLP 4 Emociones

4 Emociones					
MULTILAYER PERCEPTRON					
	FELIZ	TRISTE	SORPRESA	RABIA	TOTAL
FELIZ	88,89%	2,78%	8,33%	0,00%	100%
TRISTE	8,33%	63,89%	5,56%	22,22%	100%
SORPRESA	2,78%	5,56%	86,11%	5,56%	100%
RABIA	5,56%	22,22%	2,78%	69,44%	100%
TOTAL	105,56%	94,44%	102,78%	97,22%	

En la Tabla 3.6 las precisiones para cada sentimiento disminuyen ya que aumenta el número de emociones. Este clasificador ha tenido más dificultades para detectar triste y rabia. Son dos expresiones muy similares ya que el ser humano suele mover los mismos músculos para estas dos emociones, cosa que dificulta su detección.

Tabla 3.7 Matriz de confusión MLP 7 Emociones

	7 Emociones							TOTAL
	MULTILAYER PERCEPTRON							
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	50,00%	27,78%	5,56%	2,78%	5,56%	0,00%	8,33%	100%
FELIZ	22,22%	41,67%	2,78%	5,56%	11,11%	5,56%	11,11%	100%
TRISTE	8,33%	0,00%	83,33%	5,56%	0,00%	2,78%	0,00%	100%
SORPRESA	8,33%	2,78%	2,78%	50,00%	0,00%	22,22%	13,89%	100%
RABIA	13,89%	16,67%	0,00%	2,78%	36,11%	0,00%	30,56%	100%
DISGUSTO	2,78%	2,78%	2,78%	13,89%	0,00%	77,78%	0,00%	100%
MIEDO	13,89%	13,89%	0,00%	11,11%	27,78%	0,00%	33,33%	100%
TOTAL	119,44%	105,56%	97,22%	91,67%	80,56%	108,33%	97,22%	

En la Tabla 3.7 la precisión ha disminuido con 7 emociones. La precisión para detectar la emoción feliz y sorpresa ha disminuido por debajo de un 50% mientras que de triste ha aumentado. El sentimiento de miedo es el más costoso para identificar junto al sentimiento de rabia. Como ya se ha comentado anteriormente, esto es debido a que algunos sentimientos son muy parecidos en cuanto a posición de los puntos faciales. Con feliz y neutral el clasificador tiende a errar bastante, pero también hay que tener en cuenta el factor humano que, aunque los puntos faciales estarán en posiciones absolutas parecidas, no serán exactamente iguales.

3.4. Reconocimiento de sentimientos con Deep learning

En esta segunda parte se hará el estudio del rendimiento de las técnicas de aprendizaje profundo para el reconocimiento de expresiones. En esta parte del proyecto se han utilizado más herramientas que en aprendizaje automático, incluyendo las utilizadas en el apartado anterior.

Se ha seguido una metodología igual a la utilizada de aprendizaje automático en cuanto a los pasos a seguir:

- Recolectar los datos.
- Preparar los datos.
- Elegir el modelo
- Evaluar los modelos

Aunque los puntos a estudiar son los mismos en los dos casos, no se ha empleado el mismo tiempo para cada apartado.

3.4.1. Recolectar los datos

Se utilizará la misma BBDD con el mismo número de sentimientos que en la primera parte. Son 7 sentimientos en total (neutral, feliz, triste, sorpresa, rabia, disgusto y miedo). Inicialmente se ha utilizado la misma BBDD para poder darle continuidad al proyecto. Uno de los problemas que pueden aparecer, si no utilizamos una BBDD que no sea muy extensa, es el comúnmente llamado sobreajuste. Esto ocurre cuando el modelo creado aprende patrones que no se ajustan a los datos nuevos, cuando un modelo aprende de atributos de las imágenes que no son relevantes. Supongamos que solo tenemos tres imágenes de cada sentimiento, pero de feliz solo son personas con gorra y de tristes personas sin gorra. El modelo podría aprender que las personas felices llevan gorra y las personas tristes no y eso no sería correcto.

3.4.2. Preparar los datos

Se utilizarán las 36 imágenes por cada emoción, de las cuales un 75 % para entrenar (27 imágenes) y un 25 % de las imágenes para validar (9 imágenes).

Aunque es la misma BBDD que se ha utilizado anteriormente, se ha decidido recortar las imágenes para solo obtener los rostros de las imágenes. Así se elimina todas las partes de la imagen que no proporcionan información adicional. Puede ocasionar que los modelos aprendan informaciones innecesarias.

Para poder probar los modelos hace falta etiquetar las imágenes con sus correspondientes sentimientos. En aprendizaje automático se le proporcionaba al código un Excel donde se detallaba los puntos faciales y que sentimientos, en cambio aquí se le proporcionara la imagen directamente guardada en el directorio con el nombre del sentimiento. Para validar se ha decidido hacer una partición de la BBDD para poder separar las imágenes que son para entrenar y las imágenes que son para validar. La estructura de los directorios es la siguiente:

Numero de sentimientos/

Entrenar/

/feliz
/triste
/neutral
/sorpresa
/rabia
/miedo
/disgusto

Validar/

/feliz
/triste
/neutral
/sorpresa

/rabia
/miedo
/disgusto

Se probarán 3 diferentes escenarios:

- 2 sentimientos (feliz y triste)
- 4 sentimientos (feliz, triste, rabia y sorpresa)
- 7 sentimientos (feliz, triste, rabia, sorpresa, neutral, disgusto y miedo)

No todas las imágenes tendrán el mismo tamaño. Se han ajustado todas las imágenes para que tengan un tamaño de 128x128 píxeles.

Durante la ejecución del código, se ha implementado una función que permite aplicar a las imágenes varias modificaciones, entre ellas invertir la imagen, rotar la imagen para así proporcionar más robustez a los modelos.

3.4.3. Descripción de los modelos

Como ya se ha explicado anteriormente, el aprendizaje profundo trabaja con redes neuronales, donde cada red tiene un número de capas. Se ha decidido probar con tres modelos diferentes.

- Modelo 1: una capa de convolución y una capa de agrupación.
- Modelo 2: dos capas de convolución y dos capas de agrupación.
- Modelo 3: tres capas de convolución y tres capas de agrupación.

El modelo 2 incorporará las capas del modelo 1 igualmente que el modelo 3 incorporara las capas del modelo 2.

Al clasificador entran imágenes de todos los tamaños, pero se hace un reajuste a un tamaño de 128x128 comentado anteriormente. Estas imágenes están compuestas por 3 canales a escala de grises de cada color (RGB) por lo que el tamaño de la matriz será 128x128x3.

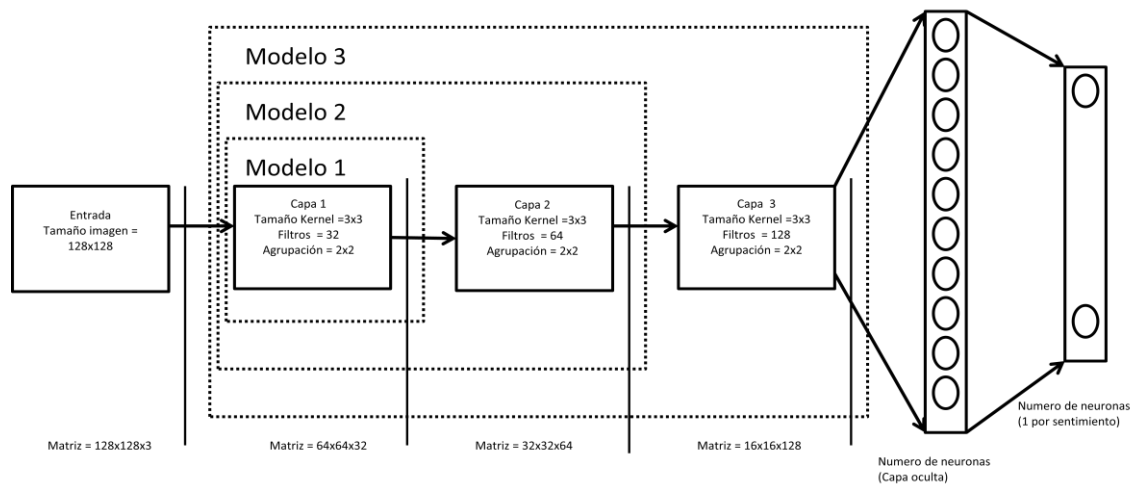


Figura 3.9 Modelos redes neuronales

En la Figura 3.9 se puede observar cómo quedaría el escenario. El número de filtros de cada modelo va aumentando. La primera capa tendrá 32 filtros, las siguientes capas duplicarán el número de filtros de la capa anterior. El tamaño del Kernel se mantiene. El número de filtros aumenta para darle más profundidad a los datos mientras que el tamaño del Kernel se ha mantenido para que afecte de la misma manera en todos los filtros.

- A la salida de la capa 1, pasaremos de 128x128x3 a 64x64x32.
- A la salida de la capa 2, pasaremos de 64x64x32 a 32x32x64.
- A la salida de la capa 3, pasaremos de 32x32x64 a 16x16x128.

Si un modelo no tiene alguna de las capas, la salida será del tamaño de la última capa aplicada.

Los tres modelos tienen al final dos capas comunes. Una capa oculta de neuronas, incorporará un total de 64 neuronas. A mayor número de neuronas en esta capa, más compleja se vuelve la red. Por último, tendrá una capa de salida donde tendrá tantas neuronas como sentimientos se desea clasificar.

3.4.4. Selección del modelo

Se seguirá un procedimiento parecido al aprendizaje automático. Primero de todo se probarán los modelos sobre dos sentimientos para poder ver cuál es el modelo de capas más eficiente. A continuación, se probarán estos modelos para 2, 4 y 7 sentimientos.

Para que la máquina pueda mejorar y vaya aprendiendo se harán varias iteraciones, llamadas épocas. Cuantas más iteraciones, más aprenderá de sus propios errores y mayor será la precisión. La precisión no aumentará infinitamente, sino que a partir de un número de iteraciones la precisión se

estabilizará. Se pondrá un número de iteraciones lo suficientemente grande como para ver como tiende a estabilizarse la precisión.

El código se ejecuta en dos fases. Una primera fase de entrenamiento donde se entrenará sobre 27 imágenes y se probará la precisión obtenida sobre esas mismas 27 imágenes, y una fase de validación donde una vez entrenado sobre las 27 imágenes probará a predecir los 9 restantes.

Se mostrará para cada prueba dos gráficas, curvas de precisión y curvas de pérdidas. Las gráficas de las curvas de precisión se mostrarán como la máquina ha ido aprendiendo a lo largo de las épocas tanto en la fase de entrenamiento como en la fase de validación. Las gráficas de las curvas de pérdidas se observarán las pérdidas a lo largo de las dos fases mencionadas.

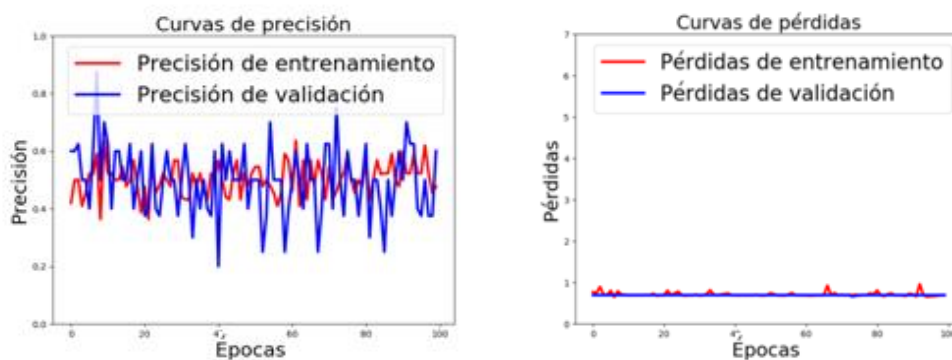


Figura 3.10 Curva de precisión y pérdidas 2 emociones modelo 1

En estas dos Figuras se observan los resultados obtenidos para dos sentimientos del modelo 1. Los resultados no han sido buenos ya que no se observa ninguna mejora. En la Figura 3.10 las curvas están alrededor de 50%. Esto significa que el modelo no ha aprendido suficiente y que la máquina hace predicciones aleatorias. En cuanto a las curvas se mantienen en cero durante todas las épocas por lo que no son datos concluyentes.

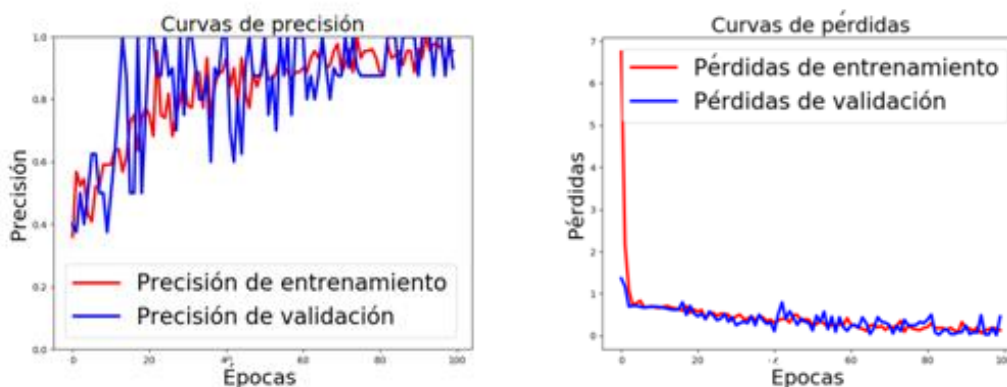


Figura 3.11 Curva de precisión y pérdidas 2 emociones modelo 2

En el modelo 2 obtenemos unos buenos resultados. A diferencia del primer modelo se obtiene una precisión del 100 % a partir de la época número 19 en

algunos puntos. Las pérdidas van disminuyendo a cada época lo que es normal ya que cada vez obtenemos una mejor precisión y la curva tiende a estabilizarse.

Aunque es posible que obtengamos unos buenos resultados con este modelo, se puede observar que la precisión tiene variaciones bastante altas. Así que se proba con el siguiente modelo para poder ver si obtenemos mejores resultados.

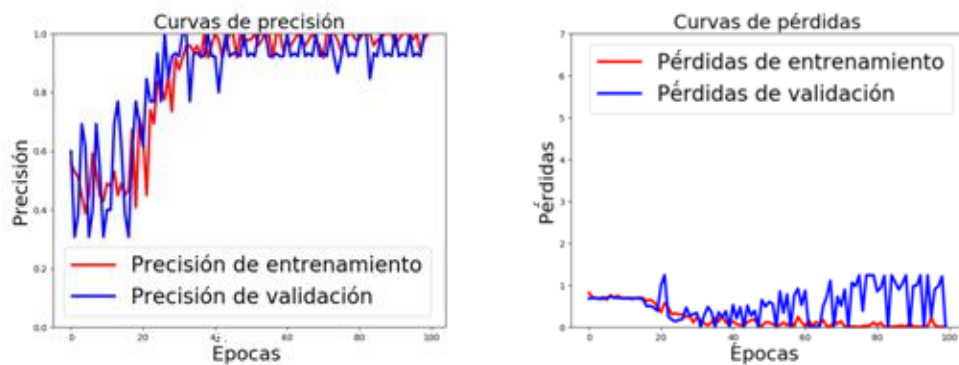


Figura 3.12 Curva de precisión y pérdidas 2 emociones modelo 3

En la Figura 3.12 se puede observar como tiende a estabilizarse la precisión ya que a partir de la época 40 la precisión alcanza un 100% tanto en la fase de entrenamiento como en la fase de validación. Las dos curvas no se separan mucho por lo que podemos decir que el modelo se ajusta bien a los datos de entrada, no se produce un sobre ajuste.

Visto los resultados obtenidos anteriormente, se utilizará el modelo 3 ya que ha sido el que ha resultado obtener mejores resultados. A partir de ahora se hará el estudio de la detección de sentimientos con el modelo 3.

3.4.5. Evaluación

Una vez testado cuál de los tres modelos funciona mejor se procederá a evaluar los diferentes sentimientos con sus correspondientes matrices de confusión.

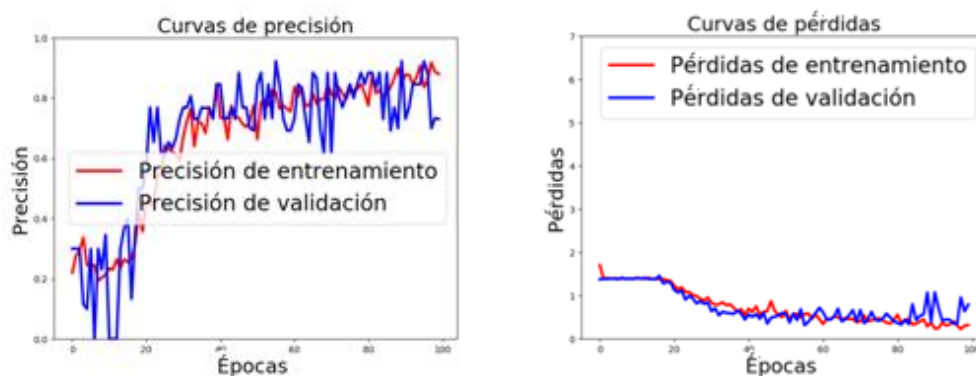
Como ya se ha podido observar en la Figura 3.12, se han obtenido unas precisiones de 100% en algunas épocas para dos sentimientos pero en la curva de pérdidas se observan aumentos y la precisión no se mantiene al 100%. Esto es debido a que una vez el modelo ha aprendido todo correctamente, todo lo que aprenda de a partir de ese momento le inducirá a error.

Tabla 3.8 Matriz de confusión 2 Emociones Modelo 3

	MODELO 3		
	FELIZ	TRISTE	TOTAL
FELIZ	88,89%	11,11%	100%
TRISTE	0,00%	100,00%	100%
TOTAL	88,89%	111,11%	

Para 2 sentimientos no se observan grandes errores. Para triste obtenemos un 100% del sentimiento de triste detectado correctamente, pero para feliz se equivoca en un 11% solamente.

Ahora se procederá a evaluar este modelo para 4 sentimientos.

**Figura 3.13** Curva de precisión y pérdidas 4 emociones modelo 3**Tabla 3.9** Matriz de confusión 2 Emociones Modelo 3

	MODELO 3				TOTAL
	FELIZ	RABIA	SORPRESA	TRISTE	
FELIZ	88,89%	0,00%	0,00%	11,11%	100%
RABIA	11,11%	55,56%	0,00%	33,33%	100%
SORPRESA	0,00%	0,00%	100,00%	0,00%	100%
TRISTE	0,00%	0,00%	0,00%	100,00%	100%
TOTAL	100,00%	55,56%	100,00%	144,44%	

En las dos gráficas anteriores se puede observar que se han obtenido muy buenos resultados. Para 4 sentimientos en algunos puntos se ha llegado a obtener una precisión del 95%, muy cerca del 100%. Las pérdidas obtenidas han ido disminuyendo a lo largo de las épocas hasta llegar a mínimos.

En la matriz de confusión se puede observar que tanto el sentimiento de sorpresa como el de tristeza han obtenido un 100% de precisión mientras que a la hora de detectar la emoción de rabia es donde se han obtenido más errores bajando a casi un 50%. Aunque la mala detección de la rabia haga bajar la precisión, se sigue obteniendo unos buenos resultados.

Este modelo también se ha ajustado muy bien a las muestras ya que las curvas de entrenamiento y de validación no se alejan, así que no ha habido sobre ajuste.

Ahora se procederá al último escenario donde se desea predecir las 7 emociones básicas. Hasta ahora se han obtenido unos buenos resultados llegando incluso a un 100 % de predicción por lo que se espera obtener unos resultados superiores a los obtenidos a aprendizaje automático.

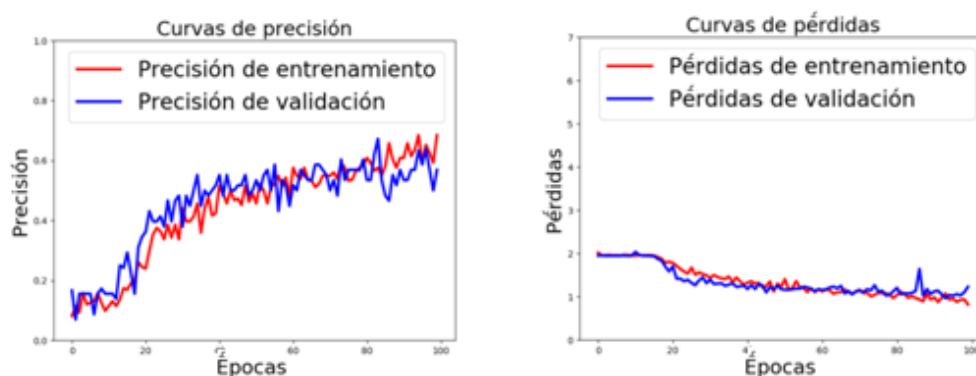


Figura 3.14 Curva de precisión y pérdidas 7 emociones modelo 3

Tabla 3.10 Matriz de confusión 7 Emociones Modelo 3

	MODELO 3							TOTAL
	DISGUSTO	FELIZ	MIEDO	NEUTRAL	RABIA	SORPRESA	TRISTE	
DISGUSTO	44,44%	11,11%	0,00%	11,11%	0,00%	0,00%	33,33%	100%
FELIZ	0,00%	88,89%	0,00%	11,11%	0,00%	0,00%	0,00%	100%
MIEDO	0,00%	0,00%	77,78%	0,00%	0,00%	22,22%	0,00%	100%
NEUTRAL	0,00%	0,00%	11,11%	44,44%	0,00%	0,00%	44,44%	100%
RABIA	22,22%	11,11%	0,00%	33,33%	11,11%	0,00%	22,22%	100%
SORPRESA	0,00%	0,00%	11,11%	0,00%	0,00%	88,89%	0,00%	100%
TRISTE	11,11%	0,00%	0,00%	33,33%	0,00%	0,00%	55,56%	100%
TOTAL	77,78%	111,11%	100,00%	133,33%	11,11%	111,11%	155,56%	

En la Figura 3.14 las pérdidas van disminuyendo a lo largo de las épocas, por lo que el modelo ha ido aprendiendo y se ha adaptado bien a las muestras. Mientras que en la otra grafica se observa que como mucho se ha obtenido una precisión de casi un 65% en algunas épocas. Se produce un ligero sobre ajuste en las últimas épocas, las curvas se separan ligeramente en las últimas épocas.

En la matriz se observa que los errores se ven dispersos por toda la matriz. Claramente se ve que la emoción que más ha errado ha sido para rabia con un 11%, mientras que las que menos ha errado han sido felices y sorpresa con un

88%. También se puede observar para el sentimiento que ha asociado mas resultados es para triste

.

3.5. Comparación

Si comparamos resultados, se ve claramente que se han obtenido mejores resultados en aprendizaje profundo que en aprendizaje automático que era lo esperado. Aun habiendo obtenido unos resultados buenos, los modelos para 7 sentimientos siguen prediciendo mal algunos sentimientos.

Tabla 3.11 Comparación de resultados finales

	Resultados finales		
	2 Emociones	4 Emociones	7 Emociones
MLP	93%	78%	56%
CNN	100%	95%	65%

En la Tabla 3.11 se muestran los mejores resultados obtenido de cada clasificador. En cualquiera de los 3 escenarios evaluados, la CNN ha obtenido mejores resultados que MLP

En aprendizaje profundo al intentar detectar 7 emociones ha perdido mucha precisión si se compara con 4 sentimientos. En aprendizaje automático aunque ha obtenido peores resultados, no ha obtenido un empeoramiento tan pronunciado al aumentar el número de sentimientos.

También se ha podido observar que el número de imágenes utilizadas para la BBDD ha sido un poco escaso. Posiblemente si aumentásemos la BBDD con más datos, los modelos tendrían más información de la cual aprender y se hubieran obtenido mejores resultados.

Si hablamos de emociones, los dos modelos les ha sido complicado detectar la emoción de rabia. Para el sentimiento de neutral, los dos han obtenido una precisión similar. En cuanto a las demás emociones un clasificador ha obtenido mejores resultados para unas y el otro para otras diferentes.

CAPÍTULO 4. CONCLUSIONES Y TRABAJO FUTURO

4.1. Conclusiones

Aprendizaje automático y aprendizaje profundo son dos técnicas que se siguen utilizando día a día y cada vez van mejorando los algoritmos de cada una. Como se ha podido observar a lo largo de este proyecto, con las dos técnicas se han podido predecir correctamente como mínimo un 50 % de las imágenes de entrada para 7 sentimientos llegando a un 65% en aprendizaje profundo.

El aprendizaje profundo utiliza técnicas de aprendizaje que necesitan una gran cantidad de datos de entrenamiento para así poder aprender todos los casos de cada clasificación. En el proyecto se ha utilizado una BBDD que, aunque se han obtenido buenos resultados, ha resultado ser escasa. Eso se puede observar a lo largo de las épocas las precisiones ya que han tenido grandes variaciones.

Otra de las grandes características que se ha podido observar es que los modelos van aprendiendo de sus propios errores lo que ayuda a que a medida que van pasando las iteraciones, la precisión va mejorando, pero no infinitamente.

Uno de los problemas que aparecen con más frecuencia en estos conceptos es el sobreajuste. Esto podía haber ocurrido ya que el número de muestras de entrada no es muy extenso. En nuestro caso no hemos observado tal problema, excepto en el último caso que aparece ligeramente. Se ha sabido adaptar bien los modelos a las muestras. A mayor número de muestras de entrenamiento, menos probabilidad hay de que los modelos sobre ajusten ya que el modelo se vuelve más robusto.

Si evaluamos el tiempo empleado para cada técnica, se podría decir que en aprendizaje automático el tiempo de aprendizaje de las técnicas no ha sido algo que haya ocupado mucho tiempo, en cambio lo que ha ocupado la gran mayoría del tiempo ha sido preparar los datos. Para aprendizaje automático se ha tenido que extraer los datos de las imágenes, preparar los datos y ordenarlos. El tiempo de ejecución del código no ha sido muy extenso.

En cambio, en aprendizaje profundo se han introducido las imágenes y es la propia máquina quien extrae los datos de las imágenes. El proceso de entendimiento y aprendizaje de las técnicas ha sido más largo e incluso el tiempo de preparación del entorno. Mirar la compatibilidad de las librerías ha sido algo complicado. El tiempo de compilación también ha sido más largo pero una vez acabado de preparar el entorno, ha sido menos complicado poder hacer cambios en el código o introducir más imágenes si se hubiese querido.

Como conclusión final, se puede ver claramente que aprendizaje profundo es mucho más eficiente. Se han obtenido mejores resultados en la mayoría de los

casos aunque se esperaban obtener unos niveles de precisión para aprendizaje profundo más elevados.

4.2. Trabajo futuro

He podido observar que este proyecto tiene posibles continuaciones. Una de ellas sería aumentar la BBDD con más ejemplos. Al añadir más imágenes aumentaría la robustez del código para que no haya tantas variaciones en la precisión de las épocas y disminuiría la probabilidad de sobreajuste. Una vez aumentado los datos de prueba y de validación, utilizar imágenes más realistas. Todas las imágenes están centradas con la misma luminosidad y con un fondo negro. Normalmente las imágenes realizadas no son siempre así.

Con los modelos utilizados se han obtenido buenos resultados, pero aun se podrían obtener mejores resultados. Así que otra posible continuidad sería ajustar los parámetros de los clasificadores para aumentar la precisión sin sobre ajustarlos.

Un ámbito al cual se podría aplicar y sería interesante es poder aplicar estas técnicas a videos. A partir de la información obtenida se podría hacer un estudio de la evolución de la persona evaluada. Por ejemplo, ver el estrés sufrido por un empleado en su lugar de trabajo.

ANEXO 1: Matrices de confusión en Machine learning

Tabla A1.1 CM 2 Emociones SVM Sin escalar

	SVM		TOTAL
	FELIZ	TRISTE	
FELIZ	80,56%	19,44%	100%
TRISTE	11,11%	88,89%	100%
TOTAL	91,67%	108,33%	

Tabla A1.2 CM 2 Emociones DT Sin escalar

	DT		TOTAL
	FELIZ	TRISTE	
FELIZ	83,33%	16,67%	100%
TRISTE	11,11%	88,89%	100%
TOTAL	94,44%	105,56%	

Tabla A1.3 CM 2 Emociones RF Sin escalar

	RF		TOTAL
	FELIZ	TRISTE	
FELIZ	88,89%	11,11%	100%
TRISTE	13,89%	86,11%	100%
TOTAL	102,78%	97,22%	

Tabla A1.4 CM 2 Emociones MLP Sin escalar

	MLP		TOTAL
	FELIZ	TRISTE	
FELIZ	80,56%	19,44%	100%
TRISTE	22,22%	77,78%	100%
TOTAL	102,78%	97,22%	

Tabla A1.5 CM 2 Emociones SVM Escalado

	SVM		TOTAL
	FELIZ	TRISTE	
FELIZ	80,56%	19,44%	100%
TRISTE	11,11%	88,89%	100%
TOTAL	91,67%	108,33%	

Tabla A1.6 CM 2 Emociones DT Escalado

DT			
	FELIZ	TRISTE	TOTAL
FELIZ	83,33%	16,67%	100%
TRISTE	11,11%	88,89%	100%
TOTAL	94,44%	105,56%	

Tabla A1.7 CM 2 Emociones RF Escalado

RF			
	FELIZ	TRISTE	TOTAL
FELIZ	88,89%	11,11%	100%
TRISTE	13,89%	86,11%	100%
TOTAL	102,78%	97,22%	

Tabla A1.8 CM 2 Emociones MLP Escalado

MLP			
	FELIZ	TRISTE	TOTAL
FELIZ	91,67%	8,33%	100%
TRISTE	2,78%	97,22%	100%
TOTAL	94,44%	105,56%	

Tabla A1.9 CM 2 Emociones SVM Estandarizado

SVM			
	FELIZ	TRISTE	TOTAL
FELIZ	91,67%	8,33%	100%
TRISTE	2,78%	97,22%	100%
TOTAL	94,44%	105,56%	

Tabla A1.10 CM 2 Emociones DT Estandarizado

DT			
	FELIZ	TRISTE	TOTAL
FELIZ	83,33%	16,67%	100%
TRISTE	2,78%	88,89%	92%
TOTAL	86,11%	105,56%	

Tabla A1.11 CM 2 Emociones RF Estandarizado

RF			
	FELIZ	TRISTE	TOTAL
FELIZ	88,89%	11,11%	100%
TRISTE	13,89%	86,11%	100%
TOTAL	102,78%	97,22%	

Tabla A1.12 CM 2 Emociones MLP Estandarizado

MLP			
	FELIZ	TRISTE	TOTAL
FELIZ	88,89%	11,11%	100%
TRISTE	5,56%	94,44%	100%
TOTAL	94,44%	105,56%	

Tabla A1.13 CM 4 Emociones SVM Sin escalar

SVM					
	FELIZ	TRISTE	SORPRESA	RABIA	TOTAL
FELIZ	88,89%	2,78%	5,56%	2,78%	100%
TRISTE	5,56%	75,00%	0,00%	19,44%	100%
SORPRESA	8,33%	5,56%	83,33%	2,78%	100%
RABIA	5,56%	22,22%	8,33%	63,89%	100%
TOTAL	108,33%	105,56%	97,22%	88,89%	

Tabla A1.14 CM 4 Emociones DT Sin escalar

DT					
	FELIZ	TRISTE	SORPRESA	RABIA	TOTAL
FELIZ	55,56%	16,67%	11,11%	16,67%	100%
TRISTE	5,56%	36,11%	13,89%	44,44%	100%
SORPRESA	11,11%	11,11%	58,33%	19,44%	100%
RABIA	11,11%	41,67%	11,11%	36,11%	100%
TOTAL	83,33%	105,56%	94,44%	116,67%	

Tabla A1.15 CM 4 Emociones RF Sin escalar

RF					
	FELIZ	TRISTE	SORPRESA	RABIA	TOTAL
FELIZ	72,22%	19,44%	8,33%	0,00%	100%
TRISTE	25,00%	63,89%	8,33%	2,78%	100%
SORPRESA	33,33%	11,11%	52,78%	2,78%	100%
RABIA	38,89%	36,11%	11,11%	13,89%	100%
TOTAL	169,44%	130,56%	80,56%	19,44%	

Tabla A1.16 CM 4 Emociones MLP Sin escalar

	MLP				TOTAL
	FELIZ	TRISTE	SORPRESA	RABIA	
FELIZ	75,00%	19,44%	5,56%	0,00%	100%
TRISTE	16,67%	61,11%	8,33%	13,89%	100%
SORPRESA	13,89%	16,67%	66,67%	2,78%	100%
RABIA	13,89%	27,78%	5,56%	52,78%	100%
TOTAL	119,44%	125,00%	86,11%	69,44%	

Tabla A1.17 CM 4 Emociones SVM Escalado

	SVM				TOTAL
	FELIZ	TRISTE	SORPRESA	RABIA	
FELIZ	77,78%	19,44%	0,00%	2,78%	100%
TRISTE	13,89%	52,78%	8,33%	25,00%	100%
SORPRESA	11,11%	8,33%	69,44%	11,11%	100%
RABIA	8,33%	44,44%	2,78%	44,44%	100%
TOTAL	111,11%	125,00%	80,56%	83,33%	

Tabla A1.18 CM 4 Emociones DT Escalado

	DT				TOTAL
	FELIZ	TRISTE	SORPRESA	RABIA	
FELIZ	66,67%	16,67%	13,89%	2,78%	100%
TRISTE	16,67%	58,33%	5,56%	19,44%	100%
SORPRESA	13,89%	11,11%	63,89%	11,11%	100%
RABIA	8,33%	27,78%	5,56%	58,33%	100%
TOTAL	105,56%	113,89%	88,89%	91,67%	

Tabla A1.19 CM 4 Emociones RF Escalado

	RF				TOTAL
	FELIZ	TRISTE	SORPRESA	RABIA	
FELIZ	77,78%	11,11%	5,56%	5,56%	100%
TRISTE	11,11%	27,78%	16,67%	44,44%	100%
SORPRESA	13,89%	0,00%	75,00%	11,11%	100%
RABIA	5,56%	16,67%	11,11%	66,67%	100%
TOTAL	108,33%	55,56%	108,33%	127,78%	

Tabla A1.20 CM 4 Emociones MLP Escalado

	MLP				
	FELIZ	TRISTE	SORPRESA	RABIA	TOTAL
FELIZ	83,33%	8,33%	5,56%	2,78%	100%
TRISTE	5,56%	66,67%	2,78%	25,00%	100%
SORPRESA	2,78%	5,56%	88,89%	2,78%	100%
RABIA	5,56%	16,67%	5,56%	72,22%	100%
TOTAL	97,22%	97,22%	102,78%	102,78%	

Tabla A1.21 CM 4 Emociones SVM Estandarizado

	SVM				
	FELIZ	TRISTE	SORPRESA	RABIA	TOTAL
FELIZ	88,89%	5,56%	2,78%	2,78%	100%
TRISTE	2,78%	83,33%	2,78%	11,11%	100%
SORPRESA	5,56%	5,56%	86,11%	2,78%	100%
RABIA	2,78%	30,56%	8,33%	58,33%	100%
TOTAL	100,00%	125,00%	100,00%	75,00%	

Tabla A1.22 CM 4 Emociones DT Estandarizado

	DT				
	FELIZ	TRISTE	SORPRESA	RABIA	TOTAL
FELIZ	66,67%	16,67%	13,89%	2,78%	100%
TRISTE	16,67%	58,33%	5,56%	19,44%	100%
SORPRESA	13,89%	11,11%	63,89%	11,11%	100%
RABIA	8,33%	27,78%	5,56%	58,33%	100%
TOTAL	105,56%	113,89%	88,89%	91,67%	

Tabla A1.23 CM 4 Emociones RF Estandarizado

	RF				
	FELIZ	TRISTE	SORPRESA	RABIA	TOTAL
FELIZ	77,78%	11,11%	5,56%	5,56%	100%
TRISTE	11,11%	27,78%	16,67%	44,44%	100%
SORPRESA	13,89%	0,00%	75,00%	11,11%	100%
RABIA	5,56%	16,67%	11,11%	66,67%	100%
TOTAL	108,33%	55,56%	108,33%	127,78%	

Tabla A1.24 CM 4 Emociones MLP Estandarizado

	MLP				TOTAL
	FELIZ	TRISTE	SORPRESA	RABIA	
FELIZ	88,89%	2,78%	8,33%	0,00%	100%
TRISTE	8,33%	63,89%	5,56%	22,22%	100%
SORPRESA	2,78%	5,56%	86,11%	5,56%	100%
RABIA	5,56%	22,22%	2,78%	69,44%	100%
TOTAL	105,56%	94,44%	102,78%	97,22%	

Tabla A1.25 CM 7 Emociones SVM Sin escalar

	SVM							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	50,00%	30,56%	2,78%	5,56%	8,33%	0,00%	2,78%	100%
FELIZ	36,11%	44,44%	0,00%	8,33%	8,33%	0,00%	2,78%	100%
TRISTE	11,11%	0,00%	77,78%	8,33%	0,00%	2,78%	0,00%	100%
SORPRESA	11,11%	8,33%	8,33%	47,22%	5,56%	16,67%	2,78%	100%
RABIA	13,89%	13,89%	0,00%	5,56%	47,22%	0,00%	19,44%	100%
DISGUSTO	0,00%	2,78%	2,78%	25,00%	0,00%	69,44%	0,00%	100%
MIEDO	8,33%	19,44%	2,78%	13,89%	27,78%	0,00%	27,78%	100%
TOTAL	130,56%	119,44%	94,44%	113,89%	97,22%	88,89%	55,56%	

Tabla A1.26 CM 7 Emociones DT Sin escalar

	DT							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	33,33%	22,22%	8,33%	8,33%	5,56%	0,00%	22,22%	100%
FELIZ	30,56%	13,89%	0,00%	11,11%	8,33%	2,78%	33,33%	100%
TRISTE	13,89%	2,78%	52,78%	8,33%	11,11%	8,33%	2,78%	100%
SORPRESA	2,78%	13,89%	8,33%	30,56%	16,67%	22,22%	5,56%	100%
RABIA	13,89%	11,11%	8,33%	16,67%	33,33%	2,78%	13,89%	100%
DISGUSTO	0,00%	8,33%	11,11%	16,67%	5,56%	50,00%	8,33%	100%
MIEDO	13,89%	8,33%	5,56%	25,00%	16,67%	5,56%	25,00%	100%
TOTAL	108,33%	80,56%	94,44%	116,67%	97,22%	91,67%	111,11%	

Tabla A1.27 CM 7 Emociones RF Sin escalar

	RF							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	22,22%	36,11%	13,89%	0,00%	25,00%	0,00%	2,78%	100%
FELIZ	33,33%	30,56%	8,33%	5,56%	8,33%	5,56%	8,33%	100%
TRISTE	0,00%	16,67%	50,00%	2,78%	22,22%	8,33%	0,00%	100%
SORPRESA	2,78%	13,89%	11,11%	25,00%	30,56%	16,67%	0,00%	100%
RABIA	2,78%	11,11%	13,89%	13,89%	55,56%	0,00%	2,78%	100%
DISGUSTO	0,00%	2,78%	8,33%	27,78%	13,89%	47,22%	0,00%	100%
MIEDO	11,11%	27,78%	11,11%	13,89%	30,56%	5,56%	0,00%	100%
TOTAL	72,22%	138,89%	116,67%	88,89%	186,11%	83,33%	13,89%	

Tabla A1.28 CM 7 Emociones MLP Sin escalar

	MLP							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	30,56%	16,67%	5,56%	22,22%	2,78%	16,67%	5,56%	100%
FELIZ	16,67%	13,89%	2,78%	11,11%	8,33%	27,78%	19,44%	100%
TRISTE	2,78%	5,56%	66,67%	16,67%	0,00%	5,56%	2,78%	100%
SORPRESA	2,78%	11,11%	5,56%	33,33%	2,78%	36,11%	8,33%	100%
RABIA	5,56%	11,11%	5,56%	8,33%	16,67%	13,89%	38,89%	100%
DISGUSTO	0,00%	5,56%	8,33%	22,22%	2,78%	44,44%	16,67%	100%
MIEDO	8,33%	11,11%	2,78%	2,78%	16,67%	19,44%	38,89%	100%
TOTAL	66,67%	75,00%	97,22%	116,67%	50,00%	163,89%	130,56%	

Tabla A1.29 CM 7 Emociones SVM Escalado

	SVM							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	36,11%	36,11%	11,11%	2,78%	13,89%	0,00%	0,00%	100%
FELIZ	30,56%	33,33%	0,00%	11,11%	16,67%	5,56%	2,78%	100%
TRISTE	8,33%	16,67%	72,22%	2,78%	0,00%	0,00%	0,00%	100%
SORPRESA	8,33%	8,33%	11,11%	30,56%	13,89%	22,22%	5,56%	100%
RABIA	8,33%	13,89%	8,33%	25,00%	41,67%	0,00%	2,78%	100%
DISGUSTO	2,78%	2,78%	5,56%	19,44%	2,78%	63,89%	2,78%	100%
MIEDO	16,67%	22,22%	5,56%	22,22%	19,44%	0,00%	13,89%	100%
TOTAL	111,11%	133,33%	113,89%	113,89%	108,33%	91,67%	27,78%	

Tabla A1.30 CM 7 Emociones DT Escalado

	DT							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	47,22%	8,33%	2,78%	8,33%	8,33%	11,11%	13,89%	100%
FELIZ	11,11%	16,67%	13,89%	13,89%	30,56%	2,78%	11,11%	100%
TRISTE	16,67%	8,33%	55,56%	11,11%	0,00%	5,56%	2,78%	100%
SORPRESA	11,11%	13,89%	16,67%	33,33%	5,56%	11,11%	8,33%	100%
RABIA	2,78%	13,89%	2,78%	25,00%	33,33%	2,78%	19,44%	100%
DISGUSTO	2,78%	11,11%	5,56%	11,11%	5,56%	55,56%	8,33%	100%
MIEDO	11,11%	22,22%	11,11%	13,89%	13,89%	0,00%	27,78%	100%
TOTAL	102,78%	94,44%	108,33%	116,67%	97,22%	88,89%	91,67%	

Tabla A1.31 CM 7 Emociones RF Escalado

	RF							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	13,89%	38,89%	33,33%	2,78%	5,56%	2,78%	2,78%	100%
FELIZ	11,11%	33,33%	11,11%	5,56%	16,67%	8,33%	13,89%	100%
TRISTE	2,78%	13,89%	75,00%	0,00%	0,00%	2,78%	5,56%	100%
SORPRESA	2,78%	8,33%	8,33%	22,22%	16,67%	27,78%	13,89%	100%
RABIA	0,00%	16,67%	8,33%	2,78%	47,22%	2,78%	22,22%	100%
DISGUSTO	0,00%	5,56%	11,11%	13,89%	2,78%	63,89%	2,78%	100%
MIEDO	8,33%	22,22%	0,00%	11,11%	25,00%	5,56%	27,78%	100%
TOTAL	38,89%	138,89%	147,22%	58,33%	113,89%	113,89%	88,89%	

Tabla A1.32 CM 7 Emociones MLP Escalado

	MLP							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	38,89%	36,11%	11,11%	2,78%	11,11%	0,00%	0,00%	100%
FELIZ	13,89%	36,11%	19,44%	2,78%	13,89%	5,56%	8,33%	100%
TRISTE	8,33%	0,00%	86,11%	5,56%	0,00%	0,00%	0,00%	100%
SORPRESA	2,78%	5,56%	5,56%	41,67%	2,78%	30,56%	11,11%	100%
RABIA	5,56%	16,67%	0,00%	0,00%	52,78%	0,00%	25,00%	100%
DISGUSTO	0,00%	2,78%	2,78%	36,11%	0,00%	58,33%	0,00%	100%
MIEDO	8,33%	8,33%	0,00%	11,11%	41,67%	2,78%	27,78%	100%
TOTAL	77,78%	105,56%	125,00%	100,00%	122,22%	97,22%	72,22%	

Tabla A1.33 CM 7 Emociones SVM Estandarizado

	SVM							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	50,00%	30,56%	2,78%	5,56%	11,11%	0,00%	0,00%	100%
FELIZ	25,00%	41,67%	0,00%	8,33%	19,44%	2,78%	2,78%	100%
TRISTE	11,11%	0,00%	80,56%	5,56%	0,00%	2,78%	0,00%	100%
SORPRESA	5,56%	13,89%	2,78%	55,56%	5,56%	13,89%	2,78%	100%
RABIA	11,11%	5,56%	2,78%	5,56%	47,22%	0,00%	27,78%	100%
DISGUSTO	0,00%	2,78%	2,78%	27,78%	0,00%	66,67%	0,00%	100%
MIEDO	5,56%	13,89%	2,78%	11,11%	36,11%	0,00%	30,56%	100%
TOTAL	108,33%	108,33%	94,44%	119,44%	119,44%	86,11%	63,89%	

Tabla A1.34 CM 7 Emociones DT Estandarizado

	DT							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	47,22%	8,33%	2,78%	8,33%	8,33%	11,11%	13,89%	100%
FELIZ	11,11%	16,67%	13,89%	13,89%	30,56%	2,78%	11,11%	100%
TRISTE	16,67%	8,33%	55,56%	11,11%	0,00%	5,56%	2,78%	100%
SORPRESA	11,11%	13,89%	16,67%	33,33%	5,56%	11,11%	8,33%	100%
RABIA	2,78%	13,89%	2,78%	25,00%	33,33%	2,78%	19,44%	100%
DISGUSTO	2,78%	11,11%	5,56%	11,11%	5,56%	55,56%	8,33%	100%
MIEDO	11,11%	22,22%	11,11%	13,89%	13,89%	0,00%	27,78%	100%
TOTAL	102,78%	94,44%	108,33%	116,67%	97,22%	88,89%	91,67%	

Tabla A1.35 CM 7 Emociones RF Estandarizado

	RF							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	13,89%	38,89%	33,33%	2,78%	5,56%	2,78%	2,78%	100%
FELIZ	11,11%	33,33%	11,11%	5,56%	16,67%	8,33%	13,89%	100%
TRISTE	2,78%	13,89%	75,00%	0,00%	0,00%	2,78%	5,56%	100%
SORPRESA	2,78%	8,33%	8,33%	22,22%	16,67%	27,78%	13,89%	100%
RABIA	0,00%	16,67%	8,33%	2,78%	47,22%	2,78%	22,22%	100%
DISGUSTO	0,00%	5,56%	11,11%	13,89%	2,78%	63,89%	2,78%	100%
MIEDO	8,33%	22,22%	0,00%	11,11%	25,00%	5,56%	27,78%	100%
TOTAL	38,89%	138,89%	147,22%	58,33%	113,89%	113,89%	88,89%	

Tabla A1.36 CM 7 Emociones MLP Estandarizado

	MLP							TOTAL
	NEUTRAL	FELIZ	TRISTE	SORPRESA	RABIA	DISGUSTO	MIEDO	
NEUTRAL	50,00%	27,78%	5,56%	2,78%	5,56%	0,00%	8,33%	100%
FELIZ	22,22%	41,67%	2,78%	5,56%	11,11%	5,56%	11,11%	100%
TRISTE	8,33%	0,00%	83,33%	5,56%	0,00%	2,78%	0,00%	100%
SORPRESA	8,33%	2,78%	2,78%	50,00%	0,00%	22,22%	13,89%	100%
RABIA	13,89%	16,67%	0,00%	2,78%	36,11%	0,00%	30,56%	100%
DISGUSTO	2,78%	2,78%	2,78%	13,89%	0,00%	77,78%	0,00%	100%
MIEDO	13,89%	13,89%	0,00%	11,11%	27,78%	0,00%	33,33%	100%
TOTAL	119,44%	105,56%	97,22%	91,67%	80,56%	108,33%	97,22%	

ANEXO 2: Matrices de confusión en Deep learning

Tabla A2.1 CM 2 Emociones Modelo 1

MODELO 1			
	FELIZ	TRISTE	TOTAL
FELIZ	100,00%	0,00%	100%
TRISTE	100,00%	0,00%	100%
TOTAL	200,00%	0,00%	

Tabla A2.2 CM 2 Emociones Modelo 2

MODELO 3			
	FELIZ	TRISTE	TOTAL
FELIZ	88,89%	11,11%	100%
TRISTE	0,00%	100,00%	100%
TOTAL	88,89%	111,11%	

Tabla A2.3 CM 2 Emociones Modelo 3

MODELO 3			
	FELIZ	TRISTE	TOTAL
FELIZ	88,89%	11,11%	100%
TRISTE	0,00%	100,00%	100%
TOTAL	88,89%	111,11%	

Tabla A2.4 CM 4 Emociones Modelo 1

MODELO 1					
	FELIZ	RABIA	SORPRESA	TRISTE	TOTAL
FELIZ	77,78%	0,00%	0,00%	22,22%	100%
RABIA	11,11%	66,67%	11,11%	11,11%	100%
SORPRESA	0,00%	0,00%	100,00%	0,00%	100%
TRISTE	0,00%	0,00%	0,00%	100,00%	100%
TOTAL	88,89%	66,67%	111,11%	133,33%	

Tabla A2.5 CM 4 Emociones Modelo 2

MODELO 2					
	FELIZ	RABIA	SORPRESA	TRISTE	TOTAL
FELIZ	77,78%	0,00%	0,00%	22,22%	100%
RABIA	0,00%	55,56%	11,11%	33,33%	100%
SORPRESA	0,00%	0,00%	100,00%	0,00%	100%
TRISTE	0,00%	11,11%	11,11%	77,78%	100%
TOTAL	77,78%	66,67%	122,22%	133,33%	

Tabla A2.6 CM 4 Emociones Modelo 3

MODELO 3					
	FELIZ	RABIA	SORPRESA	TRISTE	TOTAL
FELIZ	88,89%	0,00%	0,00%	11,11%	100%
RABIA	11,11%	55,56%	0,00%	33,33%	100%
SORPRESA	0,00%	0,00%	100,00%	0,00%	100%
TRISTE	0,00%	0,00%	0,00%	100,00%	100%
TOTAL	100,00%	55,56%	100,00%	144,44%	

Tabla A2.7 CM 7 Emociones Modelo 1

MODELO 1								
	DISGUSTO	FELIZ	MIEDO	NEUTRAL	RABIA	SORPRESA	TRISTE	TOTAL
DISGUSTO	11,11%	22,22%	0,00%	11,11%	11,11%	0,00%	44,44%	100%
FELIZ	0,00%	77,78%	11,11%	11,11%	0,00%	0,00%	0,00%	100%
MIEDO	0,00%	0,00%	44,44%	0,00%	11,11%	44,44%	0,00%	100%
NEUTRAL	0,00%	0,00%	0,00%	11,11%	11,11%	22,22%	55,56%	100%
RABIA	0,00%	22,22%	0,00%	22,22%	33,33%	11,11%	11,11%	100%
SORPRESA	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	100%
TRISTE	0,00%	0,00%	11,11%	22,22%	0,00%	0,00%	66,67%	100%
TOTAL	11,11%	122,22%	66,67%	77,78%	66,67%	177,78%	177,78%	

Tabla A2.8 CM 7 Emociones Modelo 2

MODELO 2								
	DISGUSTO	FELIZ	MIEDO	NEUTRAL	RABIA	SORPRESA	TRISTE	TOTAL
DISGUSTO	44,44%	11,11%	0,00%	22,22%	11,11%	0,00%	11,11%	100%
FELIZ	0,00%	77,78%	0,00%	11,11%	0,00%	0,00%	11,11%	100%
MIEDO	0,00%	0,00%	33,33%	0,00%	11,11%	55,56%	0,00%	100%
NEUTRAL	0,00%	0,00%	22,22%	44,44%	0,00%	0,00%	33,33%	100%
RABIA	33,33%	0,00%	0,00%	33,33%	11,11%	0,00%	22,22%	100%
SORPRESA	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	100%
TRISTE	11,11%	0,00%	0,00%	33,33%	0,00%	0,00%	55,56%	100%
TOTAL	88,89%	88,89%	55,56%	144,44%	33,33%	155,56%	133,33%	

Tabla A2.9 CM 7 Emociones Modelo 3

MODELO 3								
	DISGUSTO	FELIZ	MIEDO	NEUTRAL	RABIA	SORPRESA	TRISTE	TOTAL
DISGUSTO	44,44%	11,11%	0,00%	11,11%	0,00%	0,00%	33,33%	100%
FELIZ	0,00%	88,89%	0,00%	11,11%	0,00%	0,00%	0,00%	100%
MIEDO	0,00%	0,00%	77,78%	0,00%	0,00%	22,22%	0,00%	100%
NEUTRAL	0,00%	0,00%	11,11%	44,44%	0,00%	0,00%	44,44%	100%
RABIA	22,22%	11,11%	0,00%	33,33%	11,11%	0,00%	22,22%	100%
SORPRESA	0,00%	0,00%	11,11%	0,00%	0,00%	88,89%	0,00%	100%
TRISTE	11,11%	0,00%	0,00%	33,33%	0,00%	0,00%	55,56%	100%
TOTAL	77,78%	111,11%	100,00%	133,33%	11,11%	111,11%	155,56%	

ANEXO 3: Código Machine learning

```
import pandas as pd
from sklearn import svm
from sklearn.model_selection import cross_val_score
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.datasets import make_classification
from sklearn.model_selection import cross_val_predict
from sklearn.metrics import confusion_matrix
from sklearn.neural_network import MLPClassifier
from sklearn.preprocessing import StandardScaler

#Aquí se cargan los datos de entrada
imagenes=pd.read_csv('C:/Users/adris/PycharmProjects/Facial
Detection/image/BBDD/BBDD_7_sentimientos_sinbarbilla_normalizado.csv',
header=None)
emociones=pd.read_csv('C:/Users/adris/PycharmProjects/Facial
Detection/image/BBDD/emociones_7_sentimientos.csv',header=None)
scaler = StandardScaler(copy=True, with_mean=True, with_std=True)
imagenes=scaler.fit_transform(imagenes)

#Codigo clasificador SVM
clf =svm.SVC(kernel='linear',C=1)
scores= cross_val_score(clf,imagenes,emociones,cv=10)
print("Accuracy: %0.2f (+/- %0.2f)" % (scores.mean(), scores.std()*2))
pred= cross_val_predict(clf, imagenes, emociones, cv=10)
conf_mat = confusion_matrix(emociones, pred)
print conf_mat

#Codigo clasificador DT
clf=DecisionTreeClassifier(random_state=0)
scores= cross_val_score(clf,imagenes,emociones,cv=10)
print("Accuracy: %0.2f (+/- %0.2f)" % (scores.mean(), scores.std()*2))
pred= cross_val_predict(clf, imagenes, emociones, cv=10)
conf_mat = confusion_matrix(emociones, pred)
print conf_mat

#Codigo clasificador RF
clf = RandomForestClassifier(n_estimators=100,
max_depth=2,random_state=0)
scores= cross_val_score(clf,imagenes,emociones,cv=10)
print("Accuracy: %0.2f (+/- %0.2f)" % (scores.mean(), scores.std()*2))
pred= cross_val_predict(clf, imagenes, emociones, cv=10)
conf_mat = confusion_matrix(emociones, pred)
print conf_mat

#Codigo clasificador MLP
clf = MLPClassifier(solver='lbfgs')
scores= cross_val_score(clf,imagenes,emociones,cv=10)
print("Accuracy: %0.2f (+/- %0.2f)" % (scores.mean(), scores.std()*2))
pred= cross_val_predict(clf, imagenes, emociones, cv=10)
conf_mat = confusion_matrix(emociones, pred)
print conf_ma
```


ANEXO 4: Código Deep learning

```

from keras.preprocessing.image import ImageDataGenerator
from keras.models import Sequential
from keras.layers import Conv2D, MaxPooling2D
from keras.layers import Activation, Dropout, Flatten, Dense
from keras import backend as K
import tensorflow as tf
import numpy as np
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
from matplotlib import *
import matplotlib.pyplot as plt

os.environ['TF_CPP_MIN_LOG_LEVEL']='2'
tf.device('/gpu:0')
sess = tf.Session(config=tf.ConfigProto(log_device_placement=True))

img_width, img_height = 128,128

#Aquí hay que poner los directorios de entrenamiento y validación
train_data_dir =
'C:/Users/adris/PycharmProjects/Keras1/data/BBDDantigua/train/train7'
validation_data_dir =
'C:/Users/adris/PycharmProjects/Keras1/data/BBDDantigua/validation/val
idation7'
nb_train_samples = 189
nb_validation_samples = 63
epochs = 100
batch_size = 5
kernel_size = 3
pool_size = 2
tamanyo_filtro = 32

if K.image_data_format() == 'channels_first':
    input_shape = (3, img_width, img_height)
else:
    input_shape = (img_width, img_height, 3)

model = Sequential()
#Primera capa oculta de 32 filtros, kernel de 3x3 y poolsize de 2x2
model.add(Conv2D(tamanyo_filtro, (kernel_size,
kernel_size),padding='same', input_shape=input_shape))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(pool_size, pool_size),))

#Segunda capa oculta de 64 filtros, kernel de 3x3 y poolsize de 2x2
model.add(Conv2D(2*tamanyo_filtro, (kernel_size,
kernel_size),padding='same'))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(pool_size, pool_size)))

#Tercera capa oculta de 128 filtros, kernel de 3x3 y poolsize de 2x2
model.add(Conv2D(4*tamanyo_filtro, (kernel_size,
kernel_size),padding='same'))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(pool_size, pool_size)))

```

```

model.add(Flatten())
model.add(Dense(64))
model.add(Activation('relu'))
model.add(Dropout(0.5))
model.add(Dense(7))
model.add(Activation('sigmoid'))
model.summary()

model.compile(loss='categorical_crossentropy',
              optimizer='rmsprop',
              metrics=['accuracy'])

train_datagen = ImageDataGenerator(
    rescale=1. / 255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale=1. / 255)

train_generator = train_datagen.flow_from_directory(
    train_data_dir,
    target_size=(img_width, img_height),
    batch_size=batch_size,
    class_mode='categorical')

validation_generator = test_datagen.flow_from_directory(
    validation_data_dir,
    target_size=(img_width, img_height),
    batch_size=batch_size,
    class_mode='categorical',
    shuffle=False)

history=model.fit_generator(
    train_generator,
    steps_per_epoch=nb_train_samples // batch_size,
    epochs=epochs,
    validation_data=validation_generator,
    validation_steps=nb_validation_samples // batch_size)

validation_generator = test_datagen.flow_from_directory(
    validation_data_dir,
    target_size=(img_width, img_height),
    batch_size=batch_size,
    class_mode='categorical',
    shuffle=False)

Y_pred = model.predict_generator(validation_generator,
    nb_validation_samples // batch_size+1)

y_pred = np.argmax(Y_pred, axis=1)
print('Matriz de confusión')
print(confusion_matrix(validation_generator.classes, y_pred))

plt.figure(figsize=[8, 6])
plt.plot(history.history['loss'], 'r', linewidth=3.0)
plt.plot(history.history['val_loss'], 'b', linewidth=3.0)
plt.legend(['Pérdidas de entrenamiento', 'Pérdidas de validación'],
    fontsize=24)
plt.xlabel('Epocas ', fontsize=22)
plt.ylabel('Pérdidas', fontsize=22)

```

```
plt.ylim(0,7)
plt.title('Curvas de pérdidas', fontsize=22)
plt.show()

plt.figure(figsize=[8, 6])
plt.plot(history.history['acc'], 'r', linewidth=3.0)
plt.plot(history.history['val_acc'], 'b', linewidth=3.0)
plt.legend(['Precisión de entrenamiento', 'Precisión de validación'],
           fontsize=24)
plt.xlabel('Epocas ', fontsize=22)
plt.ylabel('Precisión', fontsize=22)
plt.ylim(0,1)
plt.title('Curvas de precisión', fontsize=22)
plt.show()
```

GLOSARIO

BBDD : Base de datos

CNN : Convolutional Neural Network

SVM : Support Vector Machines

DT : Decision Tree

RF : Random Forest

MLP : Multilayer Perceptron

CUDA : Computer Unified Device Architecture

GPU : Unidad de procesamiento gráfico

CM : Matriz de confusión

BIBLIOGRAFIA

- [1] González Dono, M. "Los 10 usos más comunes en Machine Learning e Inteligencia Artificial", *Blogthinking*, 2017 <https://blogthinkbig.com/los-10-usos-mas-comunes-en-machine-learning-e-inteligencia-artificial>
- [2] Redacción Prnoticias, "6 aplicaciones reales del Deep learning", *Prnoticias*, 2016 <https://prnoticias.com/comunicacion/nombramientos/20154951-6-aplicaciones-reales-del-deep-learning>
- [3] Pasillas, Andric "¿Qué es Machine learning?", *Adext*, 2018 <https://blog.adext.com/es/machine-learning-guia-completa>
- [4] Gonzalez, Andres, "Conceptos básicos de Machine learning", *Clever Data*, 2014 <https://cleverdata.io/conceptos-basicos-machine-learning/>
- [5] Zambrano, Juan, "¿Aprendizaje supervisado o no supervisado? Conoce sus diferencias dentro del machine learning y la automatización inteligente", *Medium*, 2018 <https://medium.com/@juanzambrano/aprendizaje-supervisado-o-no-supervisado-39ccf1fd6e7b>
- [6] Sancho Caparrini, Fernando, "Clasificación Supervisada y No Supervisada", *CS*, 2018 <http://www.cs.us.es/~fsancho/?e=77>
- [7] Banafa, Ahmed, "¿Qué es el aprendizaje profundo?", *OpenMind*, 2016 <https://www.bbvaopenmind.com/que-es-el-aprendizaje-profundo/>
- [8] Bagnato, Juan Ignacio, "Aprendizaje Profundo: una Guía rápida", *Aprende Machine Learning*, 2017 <http://www.aprendemachinelearning.com/aprendizaje-profundo-una-guia-rapida/>
- [9] Bagnato, Juan Ignacio, "Una sencilla Red Neuronal en Python con Keras y Tensorflow", *Aprende Machine Learning*, 2018 <http://www.aprendemachinelearning.com/una-sencilla-red-neuronal-en-python-con-keras-y-tensorflow/>
- [10] Rouse, Margaret, "Aprendizaje profundo", *Search Data Center*, 2017 <https://searchdatacenter.techtarget.com/es/definicion/Aprendizaje-profundo-deep-learning>
- [11] Calvo, Diego, "Red Neuronal Convolutacional CNN", *Diego Calvo*, 2017 <http://www.diegocalvo.es/red-neuronal-convolutacional-cnn/>
- [12] Torres, Jordi, "First contact with Deep learning", *Jordi Torres. Ai*, 2018 <https://torres.ai/artificial-intelligence-content/first-contact-deep-learning-practical-introduction-keras/>
- [13] "Pycharm", *Wikipedia*, 2018 <https://en.wikipedia.org/wiki/PyCharm>

- [14] "Numpy", *Wikipedia*, 2018 <https://es.wikipedia.org/wiki/NumPy>
- [15] "Open CV", *Wikipedia*, 2018 <https://es.wikipedia.org/wiki/OpenCV>
- [16] "CUDA", *Wikipedia*, 2018 <https://es.wikipedia.org/wiki/CUDA>
- [17] "CUDNN", *NVIDIA*, 2018 <https://developer.nvidia.com/cudnn>
- [18] Bagnato, Juan Ignacio, "7 pasos del Machine Learning para construir tu máquina", *Aprende Machine Learning*, 2018
<http://www.aprendemachinelearning.com/7-pasos-machine-learning-construir-maquina/>
- [19] Rosebrock, Adrian, "Detect eyes, nose, lips, and jaw with dlib, OpenCV, and Python", *PYimagesearch*, 2017
<https://www.pyimagesearch.com/2017/04/10/detect-eyes-nose-lips-jaw-dlib-opencv-python/>
- [20] "Preprocessin data", *Scikit-learn*, 2018 <http://scikit-learn.org/stable/modules/preprocessing.html>
- [21] "Máquina de vectores de soporte", *Wikipedia*, 2018
https://es.wikipedia.org/wiki/M%C3%A1quinas_de_vectores_de_soporte
- [22] Jaggi, Martin, "Geometry os Support Vector Machines", *M8j*, 2018
[http://www.m8j.net/\(All\)Geometry%20of%20Support%20Vector%20Machines](http://www.m8j.net/(All)Geometry%20of%20Support%20Vector%20Machines)
- [23] "Decision Trees o Árboles de decisión", *Unipython*, 2018
<https://www.aprenderpython.net/decision-trees-arboles-decision/>
- [24] Donges, Niklas, "The Random Forest Algorithm", *Towards data science*, 2018 <https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd>
- [25] "Random Forest Classifier", *Scikit-learn*, 2018
<http://scikitlearn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>
- [26] "Perceptrón multicapa", *Wikipedia*, 2018
https://es.wikipedia.org/wiki/Perceptr%C3%B3n_multicapa
- [27] "Neural network models", *Scikit-learn*, 2018 https://scikit-learn.org/stable/modules/neural_networks_supervised.html

[28] “Validación cruzada”, *Wikipedia*, 2018

https://es.wikipedia.org/wiki/Validaci%C3%B3n_cruzada